Computers & Industrial Engineering 171 (2022) 108383

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

A unified Maximum Entropy Principle approach for a large class of routing problems

Mayank Baranwal [a], Lavanya Marla [b,*], Carolyn Beck [b], Srinivasa M. Salapaka [c]

[a] Tata Consultancy Services Research and Indian Institute of Technology Bombay, India
[b] Department of Industrial and Enterprise Systems Engineering, University of Illinois at Urbana-Champaign, United States of America
[c] Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, United States of America

ARTICLE INFO

Keywords:
VRP
Close-enough traveling salesman problem
Unified heuristics
Statistical physics
Maximum Entropy Principle

ABSTRACT

We present a novel Maximum Entropy Principle (MEP)-based modeling and algorithmic approach, for a large class of routing and scheduling problems including the Capacitated Vehicle Routing Problem (CVRP), the Vehicle Routing Problem with soft time-windows (VRPTW) and the Close-Enough Traveling Salesman Problem (CETSP). The MEP models routing and scheduling as 'equivalent' partitioning or clustering problems with side-constraints, and employs tools from statistical physics for assigning resources (routes/vehicles) to each node such that the resource allocation results in feasible, sub-optimal routes. The MEP can flexibly incorporate side-constraints related to minimum tour-lengths, capacities, schedules and reachability (like CETSP). Analytically, our model results in a second-order non-linear system of complex implicit equations. We show that an iterative approach effectively solves these equations, is equivalent to a gradient descent and converges to a local minimum. Despite the non-linear optimization model, the algorithm converges to an integer optimal solution. Computationally, we compare our approach to Simulated Annealing (SA), the CMT-14 benchmarks for VRP and benchmarks for CETSP. Our approach consistently outperforms SA for multiple variants of routing problems, specifically, the CVRP, VRPTW and CETSP. On the CMT-14 benchmark instances, our approach finds the optimal (when verifiable) number of vehicles, with a cumulative tour distance within 6.2% on average, and in comparable computation times of the best-known solutions (over all approaches for each instance). We also demonstrate the efficacy of our approach on benchmark instances of the CETSP and discuss our results. This demonstrates the potential of our MEP approach to be further embedded into hybridization heuristics for further improved results.

1. Introduction

Recent emphasis on building smart urban infrastructures has motivated the development of new and efficient paradigms of power and communication networks, transportation systems and supply chain integration in cities. Disruptive urban service innovations such as Uber, Lyft, Grubhub, UberEats, and the potential of modes such as drones and UAVs in cities and warehouses, all involve route planning and scheduling. Due to the on-demand and large-scale nature of services, such problems have to be solved repeatedly and at large scales. We consider problems at the core of these service delivery settings — the Traveling Salesman Problem (TSP), the TSP with time-windows, the Vehicle Routing Problem (VRP), the Close-Enough Traveling Salesman Problem (CETSP) and their numerous variants. Applications of these important and impactful problems also extend widely, including to printed circuit boards (Matai, Singh, & Mittal, 2010), overhauling gas turbine engines (Plante, Lowe, & Chandrasekaran, 1987), X-ray crystallography (Bland & Shallcross, 1987), aerial reconnaissance (Ryan, Bailey, Moore, & Carlton, 1998), and warehouse material handling (Ratliff & Rosenthal, 1983).

Most research in vehicle routing and variants focuses on a specific variant, such as the capacitated vehicle routing problem (CVRP) or the vehicle routing problem with time-windows (VRPTW) and often, a heuristic (or an exact method) is designed to take into account the specific criteria and structure associated with that problem variant. Due to problem complexity and size, heuristics have been more successful than exact methods (Laporte, Ropke, & Vidal, 2014). More recently, a few works such as Pisinger and Ropke (2007) and Ropke and Pisinger (2006) have built more general, "out-of-the-box" unified heuristics that can solve several problem types. Because transportation, warehousing and delivery businesses need to solve multiple variants of VRP

* Corresponding author.
E-mail address: lavanyam@illinois.edu (L. Marla).

https://doi.org/10.1016/j.cie.2022.108383
Received 20 July 2021; Received in revised form 24 June 2022; Accepted 25 June 2022
Available online 2 July 2022
0360-8352/

problems, potentially all simultaneously, and in near-real-time, general heuristics that are quick and possess a common underlying structure that can be modified for multiple variants become significantly relevant. Our work in this paper focuses on one such promising heuristic approach, for the TSP, the multiple TSP (mTSP), the VRP, the close-enough TSP (CETSP) and the mCETSP (CETSP with m vehicles). Observe that CETSP and variants are particularly challenging as there are a continuum of possible edges between a pair of customer locations.

Our heuristic, which we call the Maximum Entropy Principle (MEP) heuristic, mimics approaches from statistical physics, used to explain combinatorial phenomena in nature. This is analogous to the *free energy principle*, that determines the most probable ensemble property (such as energy) over a large number of possible atomic configurations (Reichl, 1998). It is also analogous to the *Maximum Entropy Principle (MEP)* from statistical learning (Jaynes, 1957), with its solution strategy designed to avoid poor local optima. One special case of MEP algorithm, called *deterministic annealing* (DA) was initially developed for clustering and facility location problems in the context of data compression in the early 1990s (Rose, 1990). While developed for facility location or clustering, the DA has been effective for minimum distortion problems (Rose, 1998), routing in multi-agent networks (Kale & Salapaka, 2012), location optimization (Salapaka, Khalak, & Dahleh, 2003), and coverage control (Xu, Salapaka, & Beck, 2014).

In this work, we extend this MEP modeling framework towards multiple variants of vehicle routing problems. We demonstrate the versatility and flexibility of our approach in simultaneously handling multiple constraints, such as multiple vehicles, capacity constraints, scheduling constraints, and *proximity or close-enough* constraints. The central concept is to view routing (and scheduling) problems as partitioning or clustering location problems with scheduling captured as 'routing' in a different dimension (see Section 3.3.2). This unified viewpoint is enabled by introducing an *ordered* set of 'facilities' whose cardinality is initially equal to the number of customer locations. An additional set of constraints that reflect problem objectives, such as finding minimum tour-length routes are imposed on these 'facilities'. This results in each facility being allocated to a distinct customer location, and consequently the *order* of the set of facilities specifies the tour, i.e., the order in which different customer locations are visited. Additional constraints model other variants of routing problems. We next build efficient solution algorithms that can be employed to solve any of these variants, as follows. We first pose an optimization problem where the cost function reflects the appropriate minimum tour length (or other relevant objective) for a given configuration of decision variables. Then we determine a probability distribution to the space of decision variables that maximizes its Shannon entropy under the constraint that the expected value of the cost function is within a threshold value. The algorithm is iterative, where the threshold values are successively decreased (similar to an *annealing* process) as the corresponding distributions are *hardened*; that is, the distribution becomes binary, with value 1 at a specific configuration of decision variables and 0 at all other configurations. This configuration serves as a solution.

In Rose (1990), a DA algorithm was developed for solving a baseline version of TSP problems. In our work we significantly extend this methodology to a host of routing and scheduling variants. We exploit the MEP framework's ability to avoid poor local minima in a resource allocation problem, and thus the algorithm serves as a natural tool for subsequent adaptation to variants of routing problems. Specifically, we show that our approach enables solutions to multiple variants of the VRP. We also demonstrate significant improvement over heuristics such as basic simulated annealing (often by as much as 100%) and the capability for simultaneously incorporating vehicle capacity constraints, tour-length constraints on individual vehicles, topological constraints on vehicles' tours, scheduling and *close-enough* constraints.

For clarity, we first differentiate the MEP method from the well-known Simulated Annealing (SA) metaheuristic. Both approaches begin iterations from a high temperature with an initial set of solutions, and explore solutions according to a non-greedy gradient descent step. However, the MEP method (and DA that it builds on) chooses the solution in the next step deterministically, compared to SA that chooses the next solution probabilistically (Toth & Vigo, 2002). The 'probabilities' in this method are associations and not randomized, as we discuss in later sections. We will refer to our approach in this paper as the MEP approach rather than DA to distinguish from the original DA approach as well as from SA.

Our objectives are to present fundamental methodological advances for the MEP-type approaches, the theoretical foundations that make these possible, and demonstrate that these improve upon previous implementations of the DA-type or Elastic Net (EN) approaches. We also show that this approach belongs to the class of unified or flexible algorithms to which recent literature in vehicle routing and scheduling problems has been gravitating. Our work indicates the promise of this method, and its future potential to be embedded into adaptive or hyper heuristics enhanced through hybridization techniques (see Burke et al. (2010) and Pisinger and Ropke (2007)) in future work, to be competitive with best-known solutions in the literature.

The key contribution of this paper is the extension of the MEP-based approach to multiple classes of routing problems, as we describe in Section 3. Below we propose our solution approach to each of the problem types. For the sake of brevity, the analysis of our MEP-based approach to routing (and scheduling) problems is discussed in detail in the context of the Vehicle Routing Problem (VRP). Derivations of update equations for other variants follow similar lines and are omitted.

The contributions of our work are as follows.

1. *We expand the existing capabilities of MEP-type methods to larger classes of routing problems.* Past applications of DA have been for TSPs and mTSPs (Dueck & Scheur, 1990; Durbin, Szeliski, & Yuille, 1989; Rose, 1990; Vakhutinsky & Golden, 1994). In this work, we extend the MEP framework to a much broader class of VRP and CETSP problems, by incorporating a variety of capacity and topological constraints.
2. *Our approach improves algorithmically upon existing DA-type approaches in the literature for this class of problems.* An important aspect for these approaches (including the DA-based Elastic Net approach) is an appropriate way to choose hyperparameter values; poor choices for these often lead to unsatisfactory local minima (Stone, 1992). We alleviate this dependence on hyperparameter tuning seen in past works, by deriving a principled approach to parametric variations of the corresponding Lagrange multipliers.
3. *We prove analytically that our approach is efficient.* Our principled algorithmic approach can be characterized as a form of gradient descent step, allowing us to extend the EN-type approach to much broader classes of routing problems.
4. *We present computational results on benchmark instances and real-world instances.* Our computational results on a real-world dataset demonstrate that the solutions generated by our approach improve upon solutions used by the last-mile carrier who provided us data. We also compare our results with benchmark datasets to show that this class of methods can result in solutions within 6.2% on average of the best-of-best known CMT-14 instances for VRPs. We also identify problem characteristics for CETSPs, for which MEP methods can result in better solutions than other methods.

Outline

In Section 2, we discuss existing literature. In Section 3, we model and formulate the basic routing problem (TSP) and multiple variants. We introduce the basic MEP solution approach and extensions for the various variants in Section 4. We discuss computational experiments in Section 5, analyze the algorithm's performance in Section 6 and conclude in Section 7.

2. Literature review

The study of vehicle routing problems and the vast number of closely related classes of problems such as the TSP, mTSP, VRP with time-windows (VRPTW) enjoys a rich history that is too vast to fully enumerate. We refer the interested reader to the books (Labadie, Prins, & Prodhon, 2016; Toth & Vigo, 2002, 2014) for a detailed description of nearly all the literature on TSPs and VRPs; and limit our discussion to a broad overview. We discuss in some more detail the Deterministic Annealing class, which is a precursor to our MEP approach, and relevant near-enough solution methods.

Methods for TSP and VRP problems and variants are broadly classified into exact (Poggi & Uchoa, 2014) and heuristic approaches (Laporte et al., 2014). Exact methods do not always scale tractably beyond 100–200 customers, motivating efficient and reasonably fast heuristics. In this paper we restrict our discussion to heuristic approaches (Basuki, Hidayat, Aji, et al., 2019; Mastan, Balakrishnan, & Sankar Sekhar Raju, 2019; Stodola, Michenka, Nohel, & Rybanskỳ, 2020; Yang, You, Liu, & Pan, 2020; Zhang, Yang, Zhang, & Gen, 2020).

Heuristic approaches have historically begun with construction heuristics, among which the Clarke and Wright savings heuristic (Basuki et al., 2019; Clarke & Wright, 1964; Nelson, Nygard, Griffin, & Shreve, 1985; Paessens, 1988) and petal algorithms (Foster & Ryan, 1976; Renaud, Boctor, & Laporte, 1996; Ryan, Hjorring, & Glover, 1993) have stood the test of time. Among improvement heuristics, k-opt exchanges (Helsgaun, 2000; Lin & Kernighan, 1973), b-cyclic, k-transfer moves (Thompson & Psaraftis, 1993) or destroy and repair schemes (Shaw, 1997) have demonstrated the most success. The field has, over time, converged to metaheuristic-type approaches that can embed multiple heuristics and use them in combination with each other (Laporte et al., 2014). Such metaheuristics include: (i) local search methods such as simulated annealing (Osman, 1993), deterministic annealing (Dueck, 1993; Dueck & Scheur, 1990; Li, Golden, & Wasil, 2005), iterated local search (Chen, Huang, & Dong, 2010; Subramanian, Uchoa, & Ochi, 2013), variable neighborhood search (Kytojoki, Nuorito, Nuorito, Braysy, & Gendreau, 2007) and (ii) population-based heuristics such as tabu search (Zachariadis & Kiranoudis, 2010), genetic algorithms (Nagata & Braysy, 2009; Prins, 2004; Vidal, Craininc, Gendreau, Lahrichi, & Rei, 2012), ant colony algorithms (Reimann, Doerner, & Hartl, 2004), scatter search and path relinking (Tarantilis, Anagnostopoulou, & Repoussis, 2013) and learning mechanisms (Creput, Hajjam, Koukam, & Kuhn, 2012). In the past decade, these algorithms have been combined and hybridized in many ways to leverage strengths of multiple methods, amplifying their power. For example, Adaptive Large Neighborhood Search (ALNS) (Pisinger & Ropke, 2007) has emerged as an intelligent and successful way to hybridize over very large neighborhoods using multiple improvement heuristics and explore the solution space in novel ways. Other successful hybridizations have been meta-meta hybridizations (e.g. Tarantilis et al. (2013)) and combining population-based search with local search (e.g. Kytojoki et al. (2007)).

More recently, heuristic methods have expanded towards incorporating multiple variants, referred to as attributes (Vidal, Crainic, Genderau, & Prins, 2013). Fuel consumption, time-windows, heterogeneous vehicles, and time-dependent travel times are some such attributes. Such methods are referred to as unified algorithms or flexible methods. Examples include the Unified Hybrid Genetic Search (UHGS) of Vidal, Crainic, Genderau, and Prins (2014), which is highly flexible and incorporates problem-specific attributes modularly; Subramanian et al. (2013) which combines multiple depots, heterogeneous fleet and pickups and deliveries; and the ALNS of Pisinger and Ropke (2007). Vidal et al. (2014) also emphasize the importance of parameter tuning and calibration in such flexible or unified methods to achieve improved solutions by using adaptive heuristics or hyper-heuristics (Burke et al., 2010; Pisinger & Ropke, 2007).

We focus on the class of methods called DA or EN methods. The DA method (Dueck, 1993; Dueck & Scheur, 1990) or the EN method (Li et al., 2005; Vakhutinsky & Golden, 1994) used an approach based on the work in Durbin et al. (1989) on Elastic Net for the TSP. While the geometry of the EN method is governed by well-established theories from statistical physics, a fundamental drawback of these works exploring this method is the lack of a principled way to choose parameter values which often leads to unsatisfactory solutions in local minima (Stone, 1992). Their approach uses some form of Gibbs distributions, as our approach does; however, the functional form is different, and more importantly, the parameter corresponding to tour-length penalty is kept constant in their implementations. Because these flexible methods are also sensitive to hyperparameter tuning, solutions that are not comparable to other methods often result. In this paper, we alleviate this dependence on hyperparameter tuning by deriving a principled approach to parametric variations of the corresponding Lagrange multipliers, allowing us to extend the EN approach to a much broader class of VRPs, bringing it towards the class of unified algorithms. We show that as we change the annealing parameter, we need to keep updating the parameter corresponding to tour-length so that the tour-length does not abruptly change to some meaningless value. While there is literature on parameter tuning in ENs, in this work, we propose a *principled* way to optimally vary this secondary Lagrange multiplier as the annealing parameter is varied.

The close-enough routing problem has also in recent times received significant attention in the literature. Motivated by the optimal robot routing problem and wireless meter reading problems, this variant finds routes that can reach near enough to locations for the purposes of reconnaissance, meter reading, etc. The authors of Behdani and Smith (2014), Carrabs, Cerrone, Cerulli, and Gaudioso (2017), Carrabs, Cerrone, Cerulli, and Golden (2020), Gulczynski, Heath, and Price (2006), Mennell (2009) and Shuttleworth, Golden, Smith, and Wasil (2008) study this problem in detail, and in Mennell (2009) discuss the structure of the close-enough TSP and VRP problems in significant detail. The CETSP adds considerable additional challenges because there is a significant increase in the number of edges in terms of the number of points that the vehicle can touch in order to 'visit' the customer. Many formulations for this problem impose a requirement of uniform radii for all customer locations. Many existing heuristics, including those proposed in Pisinger and Ropke (2007) and Ropke and Pisinger (2006), do not address the CETSP variant in their current form. Instead, special formulations have been developed to address this problem, often with multiple phases (Dong, Yang, & Chen, 2007; Gulczynski et al., 2006; Mennell, 2009; Yuan, Orlowska, & Sadiz, 2007).

3. Problem definition and formulations

We begin by formulating the VRP and its variants with appropriate constraints to reflect routing, capacitated partitioning and scheduling components. These constrained optimization problems are combinatorially hard and thus, cannot be solved efficiently using exact methods. The MEP framework described later in Section 4 augments the various constraints to the corresponding objective functions through Lagrange multipliers. For ease of illustration, we present the VRP and its variants in increasing order of problem complexity. This is also in congruence with how the MEP framework is successively modified to address these variants. The overarching theme is to view these variants as adaptations of the basic facility location problem. The nature of coupling between these facilities is exploited in our framework to model various constraints.

Notations

Capital letters such as X, Y denote matrices; and lower case bold letters $(\mathbf{x}, \mathbf{y})$ denote column vectors. Script letters $(\mathcal{X}, \mathcal{Y})$ represent sets. $\|\mathbf{x}\|$ denotes the L_2-norm of $\mathbf{x}$. $\mathbb{R}, \mathbb{N}$ denote the set of real and natural numbers, respectively. The set of *customer* locations is denoted by $\mathcal{X} =$

Table 1
Notation used in Sections 3 and 4.

Symbol	Notation	Role
n	Number of customers	Input
m	Number of vehicles	Input
$[t_{i,\text{start}}, t_{i,\text{end}}]$	Pickup and delivery time-window for the ith customer	Input
$X = \{\mathbf{x}_i\}$	Spatial coordinates of customers	Input
Cap_i	Capacity requirement of the ith-customer	Input
$\mathcal{Y} = \{\mathbf{y_j}\}$	Set of clusters with $\mathbf{y}_j$ indicating location of centroid for jth cluster	Variable
$\mathcal{V} = \{v_{ij}\}$	Set of associations with v_{ij} indicating association b/w ith customer and jth cluster	Variable
$\{\sigma_i\}_{i=1}^n$	Permutation of customer indices {1,2,…,n} representing an ordering	Variable
$\mathcal{R}$	Set of partition indices	Variable
$D(\mathcal{Y}, \mathcal{V}, \mathcal{R})$	Cumulative distance function parameterized by $\mathcal{Y}, \mathcal{V}, \mathcal{R}$	Variable
$d(\mathbf{x}_i, \mathbf{y}_j)$	Squared Euclidean distance b/w customer location $\mathbf{x}_i$ and cluster $\mathbf{y}_j$	Variable
$P(\mathcal{Y}, \mathcal{V}, \mathcal{R})$	Probability of choosing an instance $(\mathcal{Y}, \mathcal{V}, \mathcal{R})$	Variable
$p(j\|i)$	Probability that location i is associated with cluster j	Variable
$F(\cdot)$	Free-energy functional	Variable
β	Lagrange multiplier/inverse temperature	Variable
θ	Secondary Lagrange multiplier	Variable

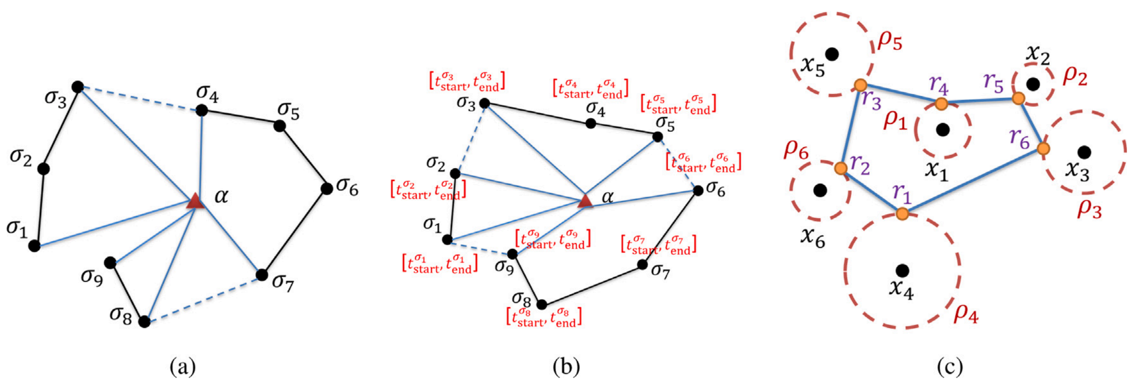


Fig. 1. Schematic of (a) Single-depot VRP with $n = 9$ customer locations, represented as black dots. (b) Single-depot VRPTW. (c) Single vehicle returning CETSP, where each location $\mathbf{x}_i$ has a radius ρ_i within which a vehicle can 'visit' the customer. The orange dots indicate the visit points $\boldsymbol{r}_j$ within the allowable radius ρ_i for each customer i.

$\{\mathbf{x}_i : 1 \leq i \leq n\}$, where $n \in \mathbb{N}$ is the number of customers. Variants such as the vehicle routing problem often require the location of a depot (or warehouse), which we denote by $\boldsymbol{\alpha}$. For variants such as the close-enough routing problems, a customer located at $\mathbf{x}_i$ is considered *visited* if the *salesman* (or vehicle) arrives anywhere within a pre-specified radius ρ_i. The distance between any two locations i and j is denoted by $d(\mathbf{x}_i, \mathbf{x}_j)$ and is assumed squared-Euclidean, i.e., $d(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|_2^2$, a choice of distance function common to many networks and vehicle routing problems. A routing sequence that visits all the customer locations (and/or depot location) exactly once is referred as a *tour*. A *tour* is defined by the sequence $(\sigma_1, \ldots, \sigma_{n+1})$ (or $(\sigma_1, \ldots, \sigma_n)$ depending upon whether or not the starting and end locations coincide), where $\sigma_i \in \{1, \ldots, n\}$ denotes the index of the ith location in the tour. Note that $\sigma_i \neq \sigma_j$ when $i \neq j$ and $1 \leq i, j \leq n$ and that $\sigma_{n+1} = \sigma_1$. Therefore each tour is fully specified by an index vector (ordered set) $\sigma = (\sigma_1, \ldots, \sigma_n)$. Note that we may refer to the agents serving a customer as salesmen or vehicles depending on the context. For the convenience of the reader, we present this and other notation that we use in this paper to describe and formulate the variants of routing problems, in Table 1. We now formally introduce the variants of routing problem in increasing order of scale and complexity.

3.1. The Traveling Salesman Problem (TSP)

We begin by introducing the most basic version of routing problems — the traveling salesman problem (TSP). Let $\mathcal{X} \triangleq \left\{\mathbf{x}_i = (x_i^{(1)}, x_i^{(2)}) : \mathbf{x}_i \in \mathbb{R}^2, 1 \leq i \leq n\right\}$ be a given set of n customer locations. The objective of the TSP (see Fig. 1a) is to find a closed tour connecting all these customer locations such that each location is visited exactly once and the total tour-length is minimized. The salesman must return to the starting customer location at the end of the tour. The TSP is mathematically formulated in the MEP framework as:

$$\min_{\sigma} \sum_{i=1}^{n} d(\mathbf{x}_{\sigma_i}, \mathbf{x}_{\sigma_{i+1}}), \quad \text{s.t.} \quad \sigma_{n+1} = \sigma_1. \tag{1}$$

By definition as described in the notations section, (1) ensures that $\sigma_l \neq \sigma_k \forall k, l \in \mathcal{X}$. Here, $d(\cdot, \cdot)$ captures the squared-Euclidean distance between two customer locations.

In practical settings, customer locations may not be known precisely, however, each customer's location may be identified in terms of a probability distribution over a set of locations. We refer to this variant of TSP by the Robust Traveling Salesman Problem (Robust-TSP). Specifically, in Robust-TSP, the location of each customer i is uncertain and lies in the set $\{\mathbf{x}_{l_i}\}$ with corresponding probability distribution $\{\zeta_{l_i}\}$ for $1 \leq l_i \leq L_i$. We present the Robust-TSP according to the MEP framework, with a single vehicle, here for clarity. The goal is to minimize the cumulative distance function given by

$$D_{\text{RTSP}}(\{\mathbf{y_j}\}, \{v_{ij}\}) = \sum_{i,j=1}^{n} v_{ij} \sum_{l_i=1}^{L_i} \zeta_{l_i} d(\mathbf{x}_{l_i}, \mathbf{y}_j) + \theta \sum_{j=1}^{n} d(\mathbf{y}_j, \mathbf{y}_{j+1}),$$
$$\text{with } \mathbf{y}_{n+1} = \mathbf{y}_1 \text{ and } \sum_{j=1}^{n} v_{ij} = 1, \tag{2}$$

where θ is the secondary Lagrange multiplier.

3.2. Close-Enough TSP (CETSP)

The CETSP occurs commonly in wireless networks, aerial drone (automated vehicle) reconnaissance, sensing networks, mission planning, and wireless electric meter reading. In a CETSP, along with the set of customer locations $\{i\}$, we are also given a corresponding set of radii

$\{\rho_i\}$ within which each location must be visited (see Fig. 1c). That is, in a CETSP a location x_i is called 'covered' if *some* point r_j in the specified circle surrounding it is visited, and not necessarily the exact location. The parameter ρ_i may reflect the range of a wireless device located at $\mathbf{x}_i$.

Due to the radius associated with each location, the CETSP does not define a specific edge (and corresponding distance) between a pair of customer locations; rather, there is a continuum of (infinite) possible edges between a pair of customer locations. Let $\{\boldsymbol{r}_j : 1 \le j \le n\}$ denote the set of locations visited by a vehicle, corresponding to the location x_i required to be 'covered'. Consequently, one can define the CETSP *tour* by the sequence of locations actually visited, $(\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_n)$. The CETSP is written as

$$\min_{\{v_{ij}\},\{\boldsymbol{r}_j\}} \sum_{j=1}^{n} \left\{ \sum_{i=1}^{n} v_{ij} d_{CE}(\mathbf{x}_i, \boldsymbol{r}_j) + d(\boldsymbol{r}_j, \boldsymbol{r}_{j+1}) \right\}; \boldsymbol{r}_{n+1} = \boldsymbol{r}_1$$

$$\text{s.t.} \quad v_{ij} \in \{0,1\}, \quad \sum_{i=1}^{n} v_{ij} = 1, \forall j; \quad \sum_{j=1}^{n} v_{ij} = 1, \forall i$$

$$\text{where} \quad d_{CE}(\mathbf{x}_i, \boldsymbol{r}_j) = \begin{cases} 0 & \text{if} \|\boldsymbol{r}_j - \mathbf{x}_i\| < \rho_i \\ \infty & \text{else} \end{cases}. \tag{3}$$

Here v_{ij} captures the association of customer location $\mathbf{x}_i$ with partition $\mathbf{r}_j$, with j indicating the order of visit. The first term in the objective function captures the cost incurred in allocating facilities to customer locations, while the second term corresponds to minimum tour-length constraint associated with the sequence $(\mathbf{r}_1, \ldots, \mathbf{r}_{n+1})$. Note that this variant also can be modeled as a multi-vehicle problem. In such a case, we would denote k as the index for the vehicle and r_{jk} as the location at which vehicle k visits the jth customer. The formulation would be modified to ensure that each customer is visited by exactly one vehicle at the point r_{jk}, and that the vehicle tours partition the customers into sets covered by each vehicle.

3.3. The Vehicle Routing Problem (VRP)

The Vehicle Routing Problem (VRP) addresses the optimal design of routes to serve a set of customers at locations $\mathcal{X} = \{\mathbf{x}_i : 1 \le i \le n\}$, using a fleet of m vehicles whose routes should begin and end at common depot. The solution is required to provide an assignment of customers to vehicles, with each customer visited once and exactly once, and to find the order in which customers are visited by each vehicle, with minimum tour cost. In a VRP, each vehicle starts at a depot location $\boldsymbol{\alpha}$, serves a subset of customers and returns to the depot. We write the objective function of the (uncapacitated) VRP mathematically in the MEP framework as

$$\min_{\substack{\sigma \\ \{\gamma_{k_j}\}}} \left\{ \sum_{i=1}^{n-1} d(\mathbf{x}_{\sigma_i}, \mathbf{x}_{\sigma_{i+1}}) + d(\mathbf{x}_{\sigma_1}, \boldsymbol{\alpha}) + d(\mathbf{x}_{\sigma_n}, \boldsymbol{\alpha}) \right.$$
$$\left. + \sum_{j=1}^{m-1} \gamma_{k_j} \left[-d(\mathbf{x}_{\sigma_{k_j}}, \mathbf{x}_{\sigma_{k_j+1}}) + d(\mathbf{x}_{\sigma_{k_j}}, \boldsymbol{\alpha}) + d(\mathbf{x}_{\sigma_{k_j+1}}, \boldsymbol{\alpha}) \right] \right\} \tag{4}$$

$$\text{s.t.} \quad \sum_{k_j} \gamma_{k_j} = 1, \quad \gamma_{k_j} \in \{0,1\}, \quad 1 \le k_j \le n-1, \forall j \tag{5}$$

where $d(\mathbf{x}_{\sigma_a}, \boldsymbol{\alpha})$ refers to the distance between customer location $\mathbf{x}_{\sigma_a}$ and depot location $\boldsymbol{\alpha}$ (see Fig. 1a). The variable γ_{k_j} takes on value 1 if the edge corresponding to the locations $(\sigma_{k_j}, \sigma_{k_j+1})$ is removed and instead, the vehicle returns from that customer to the depot. In the objective (4), the first term refers to the cost related to successively visited customers, and the second and third terms to the cost of traveling from the depot to the first customer and from the last customer back to the depot, for each vehicle; giving rise to m disjoint tours. Feasible solutions to the VRP are illustrated in Fig. 1a.

3.3.1. Capacitated VRP (CVRP)

A CVRP configures multiple vehicles to cover all customer locations, such that each customer location is visited exactly once by one vehicle/salesman and the sum of capacities of customers served in each vehicle's route is less than or equal to the maximum capacity of the vehicle. Each vehicle begins from and returns to the common depot. Let Cap_i represent the set of capacity requirements of the customers $1 \le i \le n$ and $\text{Cap}_{j,\max}$ denote the maximum carrying capacity of the vehicles $1 \le j \le m$. The CVRP can then be expressed as the capacitated version of (5) with the additional constraint that the sum of all customer capacity requirements being served by a vehicle should not exceed $\{\text{Cap}_{j,\max}\}$, i.e., if v_{ij} denotes the association of the ith-customer with the jth-vehicle, then $\sum_{i=1}^{n} v_{ij}\text{Cap}_i \le \text{Cap}_{j,\max}$ for all $1 \le j \le m$.

3.3.2. VRP with Time-Windows (VRPTW)

The VRPTW (Baranwal, Parekh, Marla, Salapaka, & Beck, 2016; Cordeau, Laporte, & Mercier, 2001; Dror, 1994) is a constrained version of the VRP, where time-windows $\{[t_{i,\text{start}}, t_{i,\text{end}}]\}$ are associated with the possible service times of each shipment i, requiring that the vehicle must reach and serve the shipment within the specified time windows (see Fig. 1b). These problems arise in city logistics, telecommunications, military applications, last mile delivery problems, liner shipping and inter-city logistics. In the MEP framework, the objective for the multi-vehicle VRPTW is formulated with *soft* penalty constraints as

$$\min_{\substack{\sigma \\ \{\gamma_{k_j}\}}} \left\{ \sum_{i=1}^{n-1} d(\mathbf{x}_{\sigma_i}, \mathbf{x}_{\sigma_{i+1}}) + d(\mathbf{x}_{\sigma_1}, \boldsymbol{\alpha}) + d(\mathbf{x}_{\sigma_n}, \boldsymbol{\alpha}) \right.$$
$$\left. + \sum_{j=1}^{m-1} \gamma_{k_j} \left[-d(\mathbf{x}_{\sigma_{k_j}}, \mathbf{x}_{\sigma_{k_j+1}}) + d(\mathbf{x}_{\sigma_{k_j}}, \boldsymbol{\alpha}) + d(\mathbf{x}_{\sigma_{k_j+1}}, \boldsymbol{\alpha}) \right] \right\} \tag{6}$$

where $\mathbf{x}_i \triangleq \left(x_i^{(1)}, \quad x_i^{(2)}, \quad 0.5\eta(t_{i,\text{start}} + t_{i,\text{end}}) \right)^T$, is a softened location of the ith customer. That is, this formulation expands the definition of the 'location' of a customer to be the spatial location as well as the mid-point of the customer's pickup and delivery time-window. The parameter η captures the relative penalty between distance-based optimization (routing) and time-window specific optimization (scheduling problems), and is described in detail for the multiple vehicles case, in Section 4.7.

4. The MEP modeling framework for the VRP and its variants

We begin by introducing the Maximum Entropy Principle (MEP) framework, its ability to find high quality solutions to non-convex/combinatorial optimization problems, and how it can be transformed to address VRP and its variants. The MEP views a typical vehicle routing problem as an abstraction of a *combination* of problems — (i) a clustering or partitioning problem based on spatial coordinates and capacities of customers, (ii) a routing problem within each cluster, and (iii) a routing problem connecting each cluster to the depot, on a Cartesian plane. This view provides the advantage of casting a large class of routing (or scheduling) problems as variants of the VRP or its core problem, the TSP. Below we first describe the MEP framework and our solution algorithm for the VRP and its variants in terms of these constituent component problems. Fig. 2 represents the overall solution methodology adopted in this work and prescribes the order in which variants of VRP with increasing complexity are successively addressed using the proposed MEP framework.

The MEP approach describes the VRP as a problem layered upon the TSP, where the loop that connects all customer locations in the TSP is partitioned into pieces and assigned to each vehicle; and each partition is connected to the depot at the beginning and end. The MEP views the vehicle routing problem as an abstraction of a *combination* of problems - (i) a clustering or partitioning problem based on spatial coordinates and capacities of customers, (ii) a routing problem within each cluster, and (iii) a routing problem connecting each cluster to the depot, on a

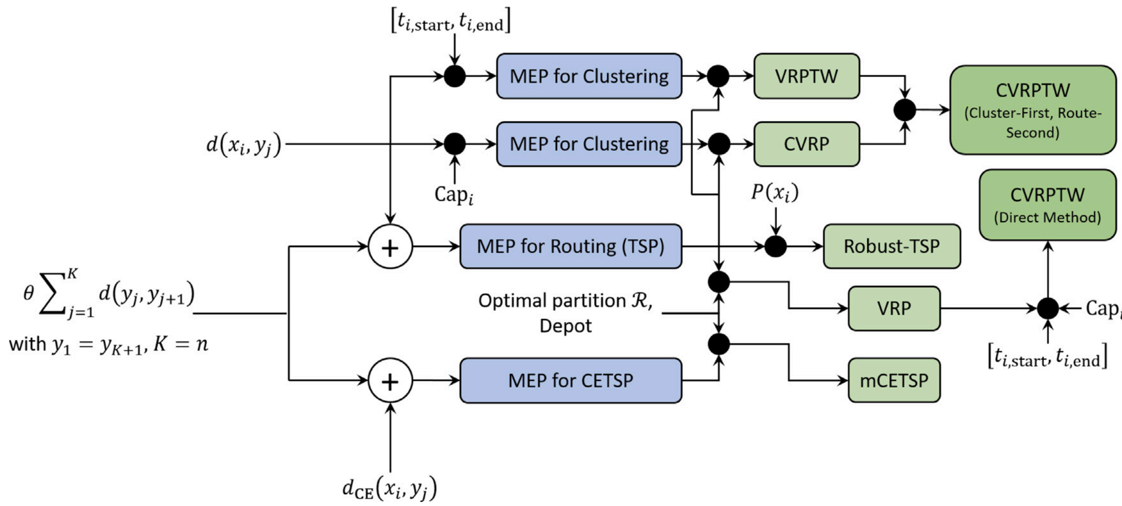


Fig. 2. Overview of MEP methodology for solving VRP and its variants — the blue colored textboxes represent the solution approach, while green colored textboxes depict the problems being solved.

Cartesian plane. This view provides the advantage of casting a large class of routing (or scheduling) problems as variants of the VRP or its core problem, the TSP. We first describe the MEP framework and our solution algorithm for the basic vehicle routing problem in terms of these constituent component problems.

For ease of exposition, we first describe our solution approach to the uncapacitated variant of the vehicle routing problem, i.e., each vehicle can serve any number of customers with the common objective of minimizing the cumulative travel distance/time over all vehicles. We then extend the approach to consider capacitated variants of the vehicle routing problem. In order to characterize a solution of an uncapacitated vehicle routing problem (henceforth referred to as VRP), we need to consider the following three elements corresponding to the three problems mentioned in the preceding paragraph: (i) a set of vehicles (clusters), (ii) associations of customers to these clusters, and (iii) a set of breakpoints that separate the clusters of each vehicle. The last element, i.e., the partition set or the set of breakpoints is used to create a fragmentation of a single route into multiple sub-routes, and is a characteristic of our proposed methodology. Let us suppose we desire to cover an entire set of customers with just three vehicles. If we first consider a single route that connects all the customers in some order and later fragment the route at indices k_1 and k_2, then the route gets partitioned into three smaller sub-routes, each considered to be served by a separate vehicle. Our goal is to obtain an optimal combination of these three elements such that the total cost of travel is minimized.

Accordingly, a configuration of a VRP is defined by the 3-tuple $(\mathcal{Y}, \mathcal{V}, \mathcal{R})$. $\mathcal{Y}$ captures the set of clusters or 'partitions' that can serve the set of customers, and $\mathcal{V}$ captures the set of associations of customers with each 'cluster' or vehicle. $\mathcal{R}$ captures the indices of facilities where one vehicle's customers stop and another vehicle's begin. Specifically, the partition set $\mathcal{R} \triangleq \{k_1, k_2, \ldots, k_{m-1}\}$ describes the set of indices where partitions occur if there are no links between clusters $\mathbf{y}_{k_l}$ and $\mathbf{y}_{k_l+1}$ for all $1 \le l \le m-1$. The objective 'distance' function for VRP associated with the instance $(\mathcal{Y}, \mathcal{V}, \mathcal{R})$ is defined as:

$$D(\mathcal{Y}, \mathcal{V}, \mathcal{R}) = D_1(\mathcal{Y}, \mathcal{V}) + D_2(\mathcal{Y}) + D_3(\mathcal{Y}, \mathcal{R}), \tag{7}$$

Here, $D_1(\mathcal{Y}, \mathcal{V})$ comprises the clustering or partitioning component (number of vehicles/facilities and associating customers to each vehicle's cluster); $D_2(\mathcal{Y})$ comprises the routing component, solved for each vehicle; and $D_3(\mathcal{Y}, \mathcal{R})$ comprises the partitions of customers corresponding to each vehicle and corrections to ensure route feasibility with the depot. We now describe in further detail each component.

4.1. MEP for the clustering or partitioning component: $D_1(\mathcal{Y}, \mathcal{V})$

Given a total of n customer locations, the partitioning problem seeks to divide them into K clusters, each allocated to one vehicle for a given set of $n \gg K$ customers, such that the *cumulative sum of distances from each customer location to the centroid of the partition is minimized*. The MEP solution approach to the partitioning problem is based on the Deterministic Annealing (DA) algorithm (Rose, 1998).

Let $\mathcal{X} = \{\mathbf{x}_i, 1 \le i \le n\}$ and $\mathcal{Y} = \{\mathbf{y}_j, 1 \le j \le K\}$ denote the sets of locations of customers and partitions, respectively. Partitioning customers into clusters is the following optimization problem (8):

$$\min_{\{\mathbf{y}_j\}} \sum_{i=1}^{n} \left\{ \min_{1\le j\le K} d(\mathbf{x}_i, \mathbf{y}_j) \right\}, \tag{8}$$

where $d(\mathbf{x}_i, \mathbf{y}_j) \in \mathbb{R}_+$ denotes the distance between the ith customer location and jth partition's centroid location.

Any solution to this problem results in clustering of the underlying domain of customers into K clusters $\{C_j\}$ such that for any point $\mathbf{x}_i \in C_j$, the nearest partition's centroid is located at $\mathbf{y}_j$. Alternatively, a partition $\{C_j\}$ can be described in terms of a set of associations $\mathcal{V} = \{v_{ij}\} \in \{0,1\}^{n\times K}$ defined as

$$v_{ij} = \begin{cases} 1 & \text{if } \mathbf{x}_i \in C_j \\ 0 & \text{else} \end{cases}.$$

Thus a *configuration* of a partitioning problem is completely described by the tuple $(\mathcal{Y}, \mathcal{V})$ with the cost for that configuration given by $D_1(\mathcal{Y}, \mathcal{V}) \triangleq \sum_{i=1}^{n} \sum_{j=1}^{K} v_{ij} d(\mathbf{x}_i, \mathbf{y}_j)$. The objective in (8) can thus be reformulated as:

$$\min_{(\mathcal{Y},\mathcal{V})} D_1(\mathcal{Y}, \mathcal{V}). \tag{9}$$

Most algorithms for partitioning or clustering problems, such as Lloyd's algorithm (Lloyd, 1982), are very sensitive to the initial assignments of partitions or facility locations and their associations with customers. This is primarily due to the distributed aspect of such problems, where any change in the location of the ith customer affects $d(\mathbf{x}_i, \mathbf{y}_j)$ only with respect to the nearest partition j. The DA algorithm (Rose, 1990, 1998), overcomes this sensitivity by allowing each location to be *partially* associated to every partition through a so-called *association probability*. Association probabilities are initially uniform, and then repeatedly updated using the maximum entropy principle. Specifically, a probability distribution $\{P(\mathcal{Y}, \mathcal{V})\}$ is ascribed over the space of combinatorial decisions, and repeatedly estimated using the

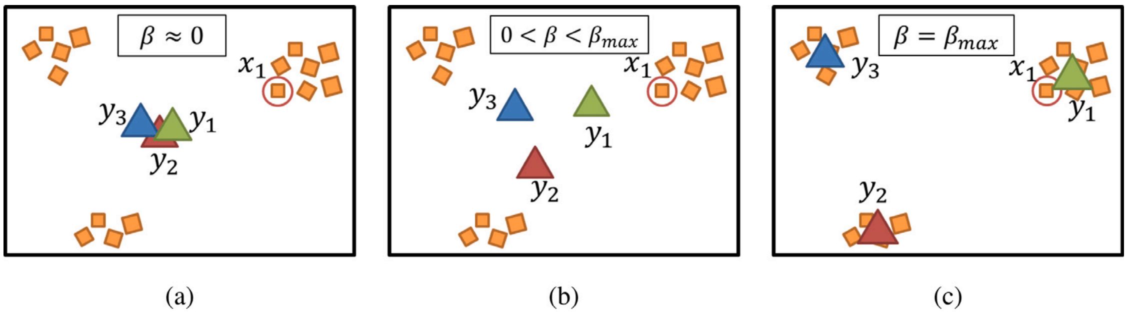


Fig. 3. The working methodology of the DA algorithm. At small values of β, the algorithm is independent of the choice of initial partitions, in fact, the partitions' centroids are optimally located at the centroid of all customer locations $\mathbf{x}_i$. At very low values of β (Figure (a)), $\mathbf{x}_1$ is uniformly associated with all the three resources - $\mathbf{y}_1$; $\mathbf{y}_2$; $\mathbf{y}_3$, i.e., $p(1|1) = p(2|1) = p(3|1) = 1/3$. As β is increased gradually, the algorithm preferentially allocates facilities based on the distribution of customer locations. At intermediate values of β (Figure (b)), the fuzziness in the associations decreases and as a result $p(1|1) > 1/3$. When $\beta \to \infty$ (Figure (c)), the resource associations become hard, i.e., $p(1|1) = 1$ and $p(2|1) = p(3|1) = 0$, resulting in hard-clustered solutions.

maximum entropy principle by solving the following related problem:

$$\max_{(\mathcal{Y},\mathcal{V})} H(P(\mathcal{Y},\mathcal{V})) \quad \text{subject to } \langle D_1(\mathcal{Y},\mathcal{V})\rangle \le D_0, \tag{10}$$

where $D = \langle D_1(\mathcal{Y},\mathcal{V})\rangle \triangleq \sum_{\mathcal{Y},\mathcal{V}} P(\mathcal{Y},\mathcal{V})D_1(\mathcal{Y},\mathcal{V})$ is the average cost and $H(P(\mathcal{Y},\mathcal{V})) \triangleq -\sum_{(\mathcal{Y},\mathcal{V})} P(\mathcal{Y},\mathcal{V})\log P(\mathcal{Y},\mathcal{V})$ is the Shannon entropy of the probability distribution $P(\mathcal{Y},\mathcal{V})$ and quantifies the randomness of the distribution. In the MEP approach, we seek to maximize entropy under the constraint that $D \le D_0,\ D_0 > 0$. The constraining parameter D_0 is successively reduced through a Lagrange multiplier.

Equivalently to (10), we instead seek to minimize its Lagrangian relaxation $\langle D\rangle - \frac{1}{\beta}H$. For a given value of β, the optimal probability distribution $P(\mathcal{Y},\mathcal{V})$ can be shown to be a Gibbs distribution, given by

$$P(\mathcal{Y},\mathcal{V}) = \frac{e^{-\beta D_1(\mathcal{Y},\mathcal{V})}}{\sum_{\mathcal{Y}',\mathcal{V}'} e^{-\beta D_1(\mathcal{Y}',\mathcal{V}')}}. \tag{11}$$

Since the partitioning problem seeks to find the clusters that maximize these probabilities, it is reasonable to consider the marginal probability distribution $\{P(\mathcal{Y})\}$, given by:

$$P(\mathcal{Y}) = \frac{e^{-\beta F_{\text{cluster}}(\mathcal{Y})}}{\sum_{\mathcal{Y}'} e^{-\beta F_{\text{cluster}}(\mathcal{Y}')}}, \tag{12}$$

where $F_{\text{cluster}}(\mathcal{Y})$ is the analog of *free energy* in statistical physics and is given by (13),

$$F_{\text{cluster}}(\mathcal{Y}) = -\frac{1}{\beta}\log Z(\mathcal{Y}) = -\frac{1}{\beta}\log \sum_{i=1}^{n}\left(\sum_{j=1}^{K} e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}\right). \tag{13}$$

If the *annealing parameter* β, is very small (≈ 0), minimizing this Lagrangian relaxation is equivalent to maximizing entropy H, which is in fact a convex problem. On the other hand, at large β, minimization of this Lagrangian relaxation is equivalent to solving the underlying problem. Thus, the Lagrange multiplier β defines a homotopy between a convex problem and the original clustering problem. In the DA algorithm discussed in Rose (1998), the Lagrangian is numerically solved for many values of β as it is increased from 0 to a large value. Fig. 3 illustrates the iterative solution process. The free energy function (13) is minimized at successively increased β values over repeated iterations (for example, Fig. 3a, b, and c).

The set $\mathcal{Y}$ of partitions that optimizes the free-energy at each β satisfies the following analytical expression:

$$\frac{\partial}{\partial \mathbf{y}_j} F_{\text{cluster}} = 0 \quad \forall j \quad \Rightarrow \sum_{i=1}^{n} p(j|i)\frac{\partial}{\partial \mathbf{y}_j} d(\mathbf{x}_i,\mathbf{y}_j) = 0 \quad \forall j, \tag{14}$$

where $p(j|i) = \dfrac{e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}}{\sum_{k=1}^{K} e^{-\beta d(\mathbf{x}_i,\mathbf{y}_k)}}$, as described earlier in (11). Computing $p(j|i)$ for each iteration (each value of β) and using it as a starting point for the next iteration is performed as shown in Rose (1990, 1998), and as shown in Fig. 3.

At low values of β, the Gibbs distribution in (11) is uniform, and the association probabilities are also uniform, as shown in Fig. 3a. This is expected since entropy is maximum when the distribution is completely random or uniform. As $\beta \to \infty$, the Gibbs distribution hardens (converges to 0 or 1), i.e., $P(\mathcal{Y})$ converges to 1 at the minimum of $F_{\text{cluster}}(\mathcal{Y})$ and 0 otherwise. Thus, the softened associations converge to hard associations as the algorithm proceeds with successively increased values of β, as seen in Fig. 3b and c. For a detailed analysis on the complexity of the basic MEP approach for partitioning, readers are encouraged to refer to Parekh, Katselis, Beck, and Salapaka (2015).

4.2. MEP for routing in each cluster: $D_2(\mathcal{Y})$

We now describe how the MEP method models the routing of vehicles in each identified cluster. Adapting the MEP algorithm for routing is not straightforward, because the original MEP method results in a partition of the underlying domain into K clusters, whereas the routing problem involves finding a *sequence* or set of sequences in which each customer location must be visited to minimize total tour length across one or more tours. Observe that if we partition the domain with as many clusters as the number of customers, i.e., if we choose $K = n$, then we expect each customer location to be uniquely co-located with one facility. The sequencing aspect is further incorporated by adding a *minimum tour-length* constraint to (10). We build upon the existing Elastic-Net (EN) based approaches, which have a fundamental drawback because of the lack of a systematic way to choose parameter values, often leading to unsatisfactory solutions (Stone, 1992). In this paper, we alleviate this dependence on hyperparameter tuning by deriving a principled approach to parametric variations of the corresponding Lagrange multipliers, thereby allowing us to broaden and extend the EN-type approaches to a much broader class of VRPs.

Rose (1990) incorporates routing for the TSP to cover all customers with a single vehicle using the minimum tour-length constraint, modifying the *free-energy* function (13) to

$$F_{\text{TSP}}(\mathcal{Y}) \triangleq F_{\text{cluster}}(\mathcal{Y}) + \theta \underbrace{\left(\sum_{j=1}^{K} d(\mathbf{y}_j,\mathbf{y}_{j+1})\right)}_{D_2(\mathcal{Y})}, \tag{15}$$

where, $\mathbf{y}_{K+1} = \mathbf{y}_1$ and θ is a secondary Lagrange multiplier associated with minimizing the tour-length obtained by joining the resources $j, j+1$ for all $1 \le j \le K$.

Fig. 4 demonstrates the working methodology of the EN approach (derived using MEP) for the basic TSP. Specifically, the desired number of partitions (depicted by triangles) are the same as the number of customer locations (depicted by squares). Fig. 4(a) shows that at very low values of β, each location is uniformly probabilistically associated with all the facilities — all the facilities are located at the centroid of the customer locations. At an intermediate value of β (Fig. 4(b)),

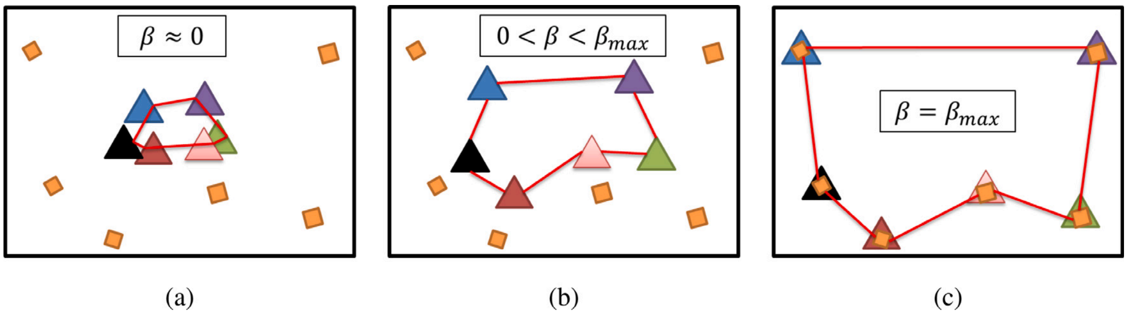


Fig. 4. Pictorial representation of the working methodology of the MEP approach for routing. At small values of β (Figure (a)), each location is uniformly associated with all facilities. At intermediate values of β (Figure (b)), the partitions develop affinity to unique customer locations, and at large values of $\beta \to \infty$ (Figure (c)), the associations of the partitions to customers locations converge to 0 or 1. For each value of β, the partitions are constrained to maintain tension in the connecting loop through the term that penalizes tour length.

the fuzziness in the associations decreases and as a result partitions start developing affinity to unique customer locations. When $\beta \to \infty$ (Fig. 4(c)), the partitions' associations with the customer locations converge to 0 or 1 (similar to 'hardness' of Gibbs distribution in case of original DA algorithm), resulting in hard-clustered solutions as desired. Observe that at each value of β, the partitions are constrained to maintain tension in the loop connecting them through the minimum tour length constraint term $D_2(\mathcal{Y})$. Furthermore, a tour is identified by the ordered set of resources $(\mathbf{y}_1, \ldots, \mathbf{y}_K)$.

Using the MEP framework, the free-energy of this system is given by:

$$F(\mathcal{Y}) = -\frac{1}{\beta}\sum_{i=1}^{n}\log\left(\sum_{j=1}^{n} e^{-\beta\sum_{l_i=1}^{L_i}\zeta_{l_i} d(\mathbf{x}_i,\mathbf{y}_j)}\right) + \theta\sum_{j=1}^{n} d(\mathbf{y}_j,\mathbf{y}_{j+1}) \tag{16}$$

At optimal location of resources $\{\mathbf{y}_j\}$, we must have

$$\frac{\partial F}{\partial \mathbf{y}_j} = 0$$

$$\Rightarrow 2\theta\left(2\mathbf{y}_j - \mathbf{y}_{j+1} - \mathbf{y}_{j-1}\right) - \frac{1}{\beta}\sum_{i=1}^{n} p(j|i)\left\{-\beta\sum_{l_i=1}^{L_i}\zeta_{l_i} 2(\mathbf{y}_j - \mathbf{x}_{l_i})\right\} = 0$$

$$\Rightarrow \mathbf{y}_j = \frac{\sum_{j=1}^{n} p(j|i)\langle\mathbf{x}_i\rangle + \theta\left(\mathbf{y}_{j+1} + \mathbf{y}_{j-1}\right)}{\sum_{j=1}^{n} p(j|i) + 2\theta}, \tag{17}$$

where $p(j|i) = \dfrac{e^{-\beta\sum_{l_i=1}^{L_i}\zeta_{l_i} d(\mathbf{x}_i,\mathbf{y}_j)}}{\sum_{k=1}^{n} e^{-\beta\sum_{l_i=1}^{L_i}\zeta_{l_i} d(\mathbf{x}_i,\mathbf{y}_k)}}$ and $\langle\mathbf{x}_i\rangle$ is the expected location of the ith-city, given by $\langle\mathbf{x}_i\rangle = \sum_{l_i=1}^{L_i}\zeta_{l_i}\mathbf{x}_{l_i}$.

4.3. Inclusion of close-enough constraints

As discussed earlier, the CETSP is computationally challenging because each location can be visited by entering a circle around the location, meaning that there is a continuum of possible edges between each pair of customer locations. Using the MEP approach, the 'resource locations' define the points of visit by a vehicle. While our approach to CETSP can be applied without loss of generality to the multiple-CETSP or close-enough-VRP, below we outline our approach for only a single vehicle, for clarity.

We introduce an additional radius parameter ρ_i, corresponding to each customer's location $\mathbf{x}_i$, which captures the distance within which the vehicle is assumed to have visited the location i. Accordingly, we modify the distance between a customer location $\mathbf{x}_i$ and partition's centroid $\mathbf{y}_j$ as:

$$d_{\mathrm{CE}}(\mathbf{x}_i,\mathbf{y}_j,\rho_i) = \left(\|\mathbf{y}_j - \mathbf{x}_i\| - \rho_i\right)^2. \tag{18}$$

As mentioned previously, the partitions' centroids $\{\mathbf{y}_j\}$ automatically define the points of visit $\mathbf{r}_j$ described in Section 3.2. The cumulative distance function corresponding to a CETSP in the proposed framework is given by:

$$D(\mathcal{Y},\mathcal{V}) = \underbrace{\sum_{i,j} v_{ij} d_{\mathrm{CE}}(\mathbf{x}_i,\mathbf{y}_j,\rho_i)}_{D_1(\mathcal{Y},\mathcal{V})} + \theta\underbrace{\sum_{j} d(\mathbf{y}_j,\mathbf{y}_{j+1})}_{D_2(\mathcal{Y})}. \tag{19}$$

Following our approach to minimization of the cumulative distance function using the maximum entropy principle, the free-energy of this system is:

$$F = -\frac{1}{\beta}\sum_{i=1}^{n}\log\Big(\sum_{j=1}^{n} e^{-\beta d_{CE}(\mathbf{x}_i,\mathbf{y}_j,\rho_i)}\Big) + \theta\sum_{j=1}^{n} d(\mathbf{y}_j,\mathbf{y}_{j+1}). \tag{20}$$

Setting the derivative of the free-energy term with respect to 'partitions' $\mathbf{y}_j$ to zero and solving for $\mathbf{y}_j$ results in the following update equation:

$$\mathbf{y}_j = \frac{\sum_{i=1}^{n} p(j|i)(\mathbf{x}_i + \rho_i\,\mathrm{sign}\,(\mathbf{y}_j - \mathbf{x}_i)) + \theta(\mathbf{y}_{j+1} + \mathbf{y}_{j-1})}{2\theta + \sum_{i=1}^{n} p(j|i)}, \tag{21}$$

where the association probabilities are now given by $p(j|i) = \left(\dfrac{e^{-\beta d_{CE}(\mathbf{x}_i,\mathbf{y}_j,\rho_i)}}{\sum_{k=1}^{n} e^{-\beta d_{CE}(\mathbf{x}_i,\mathbf{y}_k,\rho_i)}}\right)$ and $\mathrm{sign}(\cdot)$ is a vector-valued *signum* function.

4.4. MEP for connecting routes to the depot: $D_3(\mathcal{Y},\mathcal{R})$

The final component $D_3(\mathcal{Y},\mathcal{R})$, represents the connection of routes within each partition to the depot, by subtracting the distance between two successive customers in each vehicle's tour and inserting the depot $\boldsymbol{\alpha}$. This is done by first subtracting the distance between $\mathbf{y}_{k_l}$ and $\mathbf{y}_{k_l+1}$ from the original distance function, and adding links between these points to the depot.

We do this by including an additional set of decision variables to capture the indices where one vehicle's customers stop and another vehicle's begin. For example, in Fig. 5(a), the set of customers joined by continuous lines (k_1+1 until k_2) are covered by the same vehicle and the dotted lines represent the broken loop. We denote the set of customer indices where the TSP loop is partitioned, referred to as partition indices, by $\mathcal{R} = \{k_1, k_2, \ldots, k_{m-1}\}$, for a set of m vehicles. Similarly, in Fig. 5(b), for the returning mTSP, the partition indices represent the point at which the vehicle travels back to the starting point of her tour. Analogously, in Fig. 5(c), the partition indices represent the beginning and ending of each vehicle's route which are connected to the depot, forming the vehicle's route.

This component is formulated as follows:

$$D_3(\mathcal{Y},\mathcal{R}) = \theta\Bigg(d(\mathbf{y}_1,\boldsymbol{\alpha}) + d(\mathbf{y}_n,\boldsymbol{\alpha}) + \sum_{l=1}^{m-1}\Big\{-d(\mathbf{y}_{k_l},\mathbf{y}_{k_l+1}) + d(\mathbf{y}_{k_l},\boldsymbol{\alpha}) + d(\mathbf{y}_{k_l+1},\boldsymbol{\alpha})\Big\}\Bigg). \tag{22}$$

Note that here, θ is a secondary Lagrange multiplier. The objective is to seek the configuration $(\mathcal{Y},\mathcal{V},\mathcal{R})$ that minimizes the cumulative distance function in (7). $D_3(\mathcal{Y},\mathcal{R})$ not only takes into account removal of links between customers $\mathbf{y}_{k_l}$ and $\mathbf{y}_{k_l+1}$, but also adds links back to the depot $\boldsymbol{\alpha}$ to account for travel to and from the depot.

For the complete VRP, for a given instance $(\mathcal{Y},\mathcal{V},\mathcal{R})$, the cumulative distance function $D(\mathcal{Y},\mathcal{V},\mathcal{R})$ in our MEP based formulation is defined as:

$$D(\mathcal{Y},\mathcal{V},\mathcal{R}) = D_1(\mathcal{Y},\mathcal{V}) + D_2(\mathcal{Y}) + D_3(\mathcal{Y},\mathcal{R}), \tag{23}$$

4.5. Inclusion of capacity constraints on vehicles

We now discuss how to add capacity constraints to the VRP to solve the CVRP. To incorporate capacity constraints, we adopt a cluster-first route-second (Desaulniers, Desrosiers, Erdmann, Solomon, & Soumis, 2002; Gillett & Miller, 1974; Hiquebran, Alfa, Shapiro, & Gittoes, 1993; Miranda-Bront et al., 2017) approach, which is one of the most common approaches employed by several heuristics for the single-depot VRP. The MEP framework allows us to easily adapt to such an bi-level approach with clustering followed by routing. We first use the MEP approach to cluster the customer locations to account for capacity constraints (Baranwal et al., 2016), and then design economical routes over each cluster as discussed in Section 3.3. While this can be suboptimal as it is a sequential approach, we demonstrate that the solutions we achieve are competitive with existing approaches in Section 5.

Let the customers be allotted to vehicles, such that the relative capacities λ_j's of the vehicles obey,

$$p(1) : \cdots : p(j) : \cdots : p(K) = \lambda_1 : \cdots : \lambda_j : \cdots : \lambda_K.$$

In order to incorporate capacity constraints into the existing MEP formulation, p_i's are modified to capture relative weights (weighted by capacity) of each customer location. Let Cap_i be the capacity requirement for the ith-customer, then the relative weight of the ith-customer is given by $p_i = \frac{\mathrm{Cap}_i}{\sum_{i'} \mathrm{Cap}_{i'}}$. Thus, the total *mass* associated with a vehicle j is given by $p(j) = \sum_i p_i p(j|i)$, where $p(j|i)$ is the modified Gibbs distribution, i.e.,

$$p(j|i) = \frac{\eta_j e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}}{\sum_{j=1}^{K} \eta_j e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}}.$$

Here $\eta_j \in [0,1]$ specifies the relative weight of the jth vehicle. During *fuzzy* initialization (i.e., $\beta \approx 0$), η_j's are initialized to λ_j's, and thus, $p(j) = \lambda_j$ at the beginning of the annealing process. The corresponding *free-energy* function is suitably modified as:

$$F(\mathcal{Y},\eta) = -\frac{1}{\beta}\sum_{i=1}^{N} p_i \log\left(\sum_{j=1}^{K} \eta_j e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}\right). \tag{24}$$

Correspondingly, the update equation for the facility location $\mathbf{y}_j$ can be obtained by setting the derivative of the modified free-energy function w.r.t. $\mathbf{y}_j$ to zero, resulting in

$$\mathbf{y}_j = \frac{\sum_{i=1}^{N} p_i p(j|i)\mathbf{x}_i}{p(j)}, \tag{25}$$

which depends implicitly on the weight parameters η_j. Note that the desired *mass* associated with the vehicle $\mathbf{y}_j$ is λ_j, which leads to the following update rule for η_j:

$$\eta_j = \frac{\lambda_j}{\sum_{i=1}^{N} p_i \frac{e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}}{\sum_{j=1}^{K}\eta_j e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}}}. \tag{26}$$

In the capacitated version of the MEP framework, the free-energy function in (24) is deterministically optimized at successive β values by alternating between (25) and (26) until convergence. Once the customers are clustered according to the capacity, the routing within each cluster is obtained through the framework described in the beginning of this section.

4.6. Putting it all together - CVRP

The MEP framework for the CVRP proceeds iteratively, with repeated evaluations of partitions $\mathcal{Y}$ at different values of the annealing parameter β and the secondary Lagrange multiplier θ. We describe the mathematical derivations of the iterative evaluations and marginal distributions, called the update equations, through the following theorems. These update equations are embedded within the algorithm for the VRP, fully described in Algorithm 1.

Theorem 1. *Let $\mathcal{X} = \{\mathbf{x}_i, 1 \le i \le n\}$ be the set of customer locations that need to be served by m vehicles. Let $(\mathcal{Y},\mathcal{V},\mathcal{R})$ be the tuple consisting of partitions $\{\mathbf{y}_j, 1 \le j \le n\}$, set of associations $\{v_{ij}, 1 \le i,j \le n\}$ and set of partition indices $\{k_l, 1 \le l \le m-1\}$ respectively. Applying the MEP framework results in the following iterative scheme for*

(a) Partitions or clusters

$$\mathbf{y}_j = \frac{\sum_{i=1}^{n} p(j|i) + \theta\mathbf{y}_{j+1}\left(1-(m-1)\sum_{\tilde{\mathcal{R}}} Pr(j,\tilde{\mathcal{R}})\right) + \theta\mathbf{y}_{j-1}\left(1-(m-1)\sum_{\tilde{\mathcal{R}}} Pr(j-1,\tilde{\mathcal{R}})\right)}{\sum_{i=1}^{n} p(j|i) + \theta\left(2-(m-1)\sum_{\tilde{\mathcal{R}}}\left(Pr(j,\tilde{\mathcal{R}}) + Pr(j-1,\tilde{\mathcal{R}})\right)\right)} \tag{27}$$

(b) Probability of partition sets

$$Pr(k_1,\ldots,k_{m-1}) \triangleq \frac{e^{\beta\theta\sum_{l=1}^{m-1} d(\mathbf{y}_{k_l},\mathbf{y}_{k_l+1})}}{\sum_{\tilde{\mathcal{R}}} e^{\beta\theta\sum_{l=1}^{m-1} d(\mathbf{y}_{k_l},\mathbf{y}_{k_l+1})}}, \tag{28}$$

with $p(j|i)$ as defined in Section 4.1 and $\tilde{\mathcal{R}} \triangleq \mathcal{R}\setminus\{j\}$.

Proof. The cumulative distance function in a VRP is described as (23). Using the methodology proposed in (11), the corresponding Gibbs distribution is:

$$P(\mathcal{Y},\mathcal{V},\mathcal{R}) = \frac{e^{-\beta D(\mathcal{Y},\mathcal{V},\mathcal{R})}}{\sum_{\mathcal{Y}',\mathcal{V}',\mathcal{R}'} e^{-\beta D(\mathcal{Y}',\mathcal{V}',\mathcal{R}')}} \tag{29}$$

We are interested in finding the most probable set of partitions $\mathcal{Y}$ that not only minimize the cumulative tour lengths, but also provide a sequence in which a tour comprising of customer locations must be traversed by individual vehicles. Thus, we must evaluate the marginal distribution $P(\mathcal{Y})$ and aim to maximize it. Evaluation of $P(\mathcal{Y})$ requires marginalizing the numerator term in (29):

$$Z(\mathcal{Y}) = \left(\sum_{\mathcal{V}} e^{-\beta D_1(\mathcal{Y},\mathcal{V})}\right)e^{-\beta D_2(\mathcal{Y})}\left(\sum_{\mathcal{R}} e^{-\beta D_3(\mathcal{Y},\mathcal{R})}\right), \tag{30}$$

where $Z(\mathcal{Y}) = \sum_{\mathcal{V},\mathcal{R}} e^{-\beta D(\mathcal{Y},\mathcal{V},\mathcal{R})}$. If we define the *free-energy* of this system as $F(\mathcal{Y}) \triangleq -\frac{1}{\beta}\log Z(\mathcal{Y})$, then the marginal distribution $P(\mathcal{Y})$ is given by

$$P(\mathcal{Y}) = \frac{e^{-\beta F(\mathcal{Y})}}{\sum_{\mathcal{Y}'} e^{-\beta F(\mathcal{Y}')}}. \tag{31}$$

Finding the most probable set of partitions is equivalent to maximizing the marginal distribution $P(\mathcal{Y})$, which in turn is equivalent to minimizing the corresponding free-energy $F(\mathcal{Y})$. Note that

$$F(\mathcal{Y}) = \underbrace{-\frac{1}{\beta}\log\left(\sum_{\mathcal{V}} e^{-\beta D_1(\mathcal{Y},\mathcal{V})}\right)}_{F_{\text{cluster}}(\mathcal{Y})} + D_2(\mathcal{Y}) \underbrace{-\frac{1}{\beta}\log\left(\sum_{\mathcal{R}} e^{-\beta D_3(\mathcal{Y},\mathcal{R})}\right)}_{F_3(\mathcal{Y})} \tag{32}$$

At the point that minimizes the free-energy function, we must have $\frac{\partial F}{\partial \mathbf{y}_j} = 0$ for all j, where

$$\begin{aligned} F_{\text{cluster}}(\mathcal{Y}) &= -\frac{1}{\beta}\log\left(\sum_{\mathcal{V}} e^{-\beta\left\{\sum_{i=1}^{n}\sum_{j=1}^{n} v_{ij} d(\mathbf{x}_i,\mathbf{y}_j)\right\}}\right) \\ &= -\frac{1}{\beta}\log\left(\prod_{i=1}^{n}\sum_{j=1}^{n} e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}\right) \end{aligned}$$

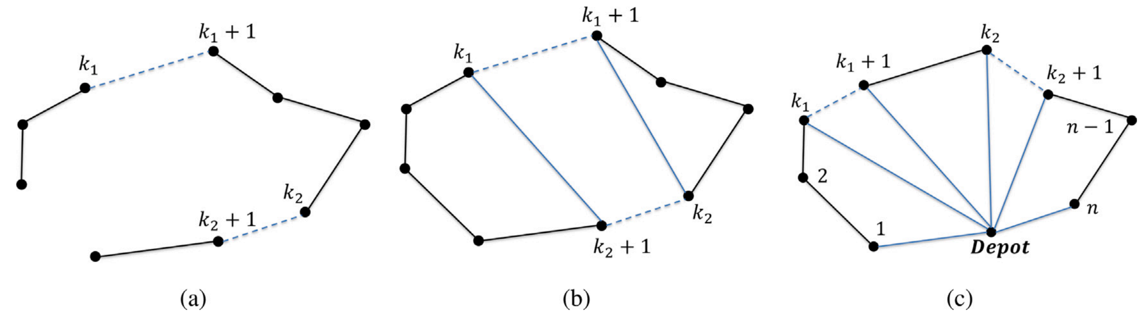


Fig. 5. Schematic of a (a) Non-returning mTSP with $m=3$ and $\mathcal{R}=\{k_1,k_2\}$, equivalent to creating three clusters and Hamiltonian paths in each cluster; (b) Returning mTSP with $m=2$ and $\mathcal{R}=\{k_1,k_2\}$, equivalent to creating two clusters and creating tours within each cluster; and (c) Single-depot VRP with $m=3$ and $\mathcal{R}=\{k_1,k_2\}$. The dashed blue lines indicate the removal of links, while solid blue lines indicate the addition of links.

$$= -\frac{1}{\beta}\sum_{i=1}^{n}\log\left(\sum_{j=1}^{n}e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}\right)$$

$$\Rightarrow \frac{\partial F_{\text{cluster}}(\mathcal{Y})}{\partial \mathbf{y}_j} = 2\sum_{i=1}^{n}p(j|i)(\mathbf{y}_j-\mathbf{x}_i), \tag{33}$$

where $p(j|i)$ is the same as defined in Section 4.1. Similarly,

$$\frac{\partial D_2(\mathcal{Y})}{\partial \mathbf{y}_j} = 2\theta(2\mathbf{y}_j-\mathbf{y}_{j+1}-\mathbf{y}_{j-1}), \tag{34}$$

where $\mathbf{y}_n=\mathbf{0}$. Finally,

$$F_3(\mathcal{Y}) = -\frac{1}{\beta}\log\left(\sum_{\mathcal{R}}e^{\beta\theta\left\{\sum_{l=1}^{m-1}d(\mathbf{y}_{k_l},\mathbf{y}_{k_l+1})\right\}}\right)$$

$$\Rightarrow \frac{\partial F_3(\mathcal{Y})}{\partial \mathbf{y}_j} = -2\theta\sum_{\mathcal{R}}\Pr(\mathcal{R})\sum_{l=1}^{m-1}\left\{(\mathbf{y}_{k_l}-\mathbf{y}_{k_l+1})\delta_{k_l,j}+(\mathbf{y}_{k_l+1}-\mathbf{y}_{k_l})\delta_{k_l,j-1}\right\}$$

$$\frac{\partial F_3(\mathcal{Y})}{\partial \mathbf{y}_j} = -2(m-1)\theta\sum_{\tilde{\mathcal{R}}}\left\{(\mathbf{y}_j-\mathbf{y}_{j+1})\Pr(j,\tilde{\mathcal{R}})\right.$$
$$\left.+(\mathbf{y}_j-\mathbf{y}_{j-1})\Pr(j-1,\tilde{\mathcal{R}})\right\}, \tag{35}$$

where $\tilde{\mathcal{R}} \triangleq \mathcal{R}\setminus\{j\}$ and $\Pr(\cdot)$ is the probability of the partition set as defined in (28).

On setting the derivative $\frac{\partial F}{\partial \mathbf{y}_j}$ to zero and collecting the terms containing $\mathbf{y}_j$, one obtains the update Eq. (27) for the partitions. ■

Corollary 1. *The update Eq. (27) for partitions is a gradient descent on the free-energy function in (32).*

Proof. Let $\{\mathbf{y}_j^{(t)}\}$ denote the set of partitions at the tth iteration of Algorithm 1. Let us represent $\mathtt{num}^{(t)}$ as the numerator and $\mathtt{den}$ as the denominator of (27). Then, we have

$$\mathbf{y}_j^{(t+1)} = \frac{\mathtt{num}^{(t)}}{\mathtt{den}}. \tag{36}$$

Note that the numerator term comprises of current iterates of partitions $\{\mathbf{y}_j^{(t)}\}$. However from (33)–(35), the derivative of free-energy function w.r.t to $\mathbf{y}_j^{(t)}$ is given by:

$$\frac{1}{2}\frac{\partial F(\mathcal{Y}^{(t)})}{\partial \mathbf{y}_j^{(t)}} = -\mathtt{num}^{(t)}+\mathbf{y}_j^{(t)}\mathtt{den}$$
$$= -\mathtt{num}^{(t)}+\left(\mathbf{y}_j^{(t)}-\mathbf{y}_j^{(t+1)}+\mathbf{y}_j^{(t+1)}\right)\mathtt{den}$$
$$\Rightarrow \frac{1}{2}\frac{\partial F(\mathcal{Y}^{(t)})}{\partial \mathbf{y}_j^{(t)}} = \left(\mathbf{y}_j^{(t)}-\mathbf{y}_j^{(t+1)}\right)\mathtt{den}, \tag{37}$$

This gives the final update equation, succinctly written as:

$$\mathbf{y}_j^{(t+1)} = \mathbf{y}_j^{(t)} - \underbrace{\frac{1}{2.\mathtt{den}}}_{\eta}\frac{\partial F(\mathcal{Y}^{(t)})}{\partial \mathbf{y}_j^{(t)}},$$

with η being the step-size of the gradient descent. ■

Algorithm 1 MEP Algorithm for CVRP

1: **input:** n, $\mathcal{X}=\{\mathbf{x}_i\}$
2: **input:** K; *//Number of vehicles*
3: **input:** $\beta_{\min}$, $\beta_{\max}$; *//Min & max value of annealing parameter*
4: **input:** $\theta_{\min}$; *//Min value of secondary Lagrange parameter*
5: **initialize:** $\beta \leftarrow \beta_{\min}$; $\mathbf{y}_j \leftarrow \frac{1}{n}\sum_{i\in n}\mathbf{x}_i$; $\theta \leftarrow \frac{5}{\sqrt{\beta_{min}}}$
6: **initialize:** $p(j|i) \leftarrow \frac{1}{n}$; initialize $\Pr(\mathcal{R})$ uniformly
7: *//Start Annealing*
8: **while** $\beta < \beta_{\max}$ **do**
9: *//Loop for controlling secondary Lagrange multiplier*
10: **while** $\theta > \theta_{\min}$ **do**
11: **while** $\mathcal{Y}$ does not converge **do**
12: $p(j|i) \leftarrow \frac{e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}}{\sum_{k\in N}e^{-\beta d(\mathbf{x}_i,\mathbf{y}_k)}}$ *//Evaluate Gibbs distribution*
13: update $\Pr(\mathcal{R})$ according to (28)
14: update $\mathcal{Y}$ according to (27) *//Find most probable set of partitions*
15: **end while**
16: *//Constrain the partitions to minimize tour-length*
17: compute length L of the tour
18: compute $F_{\text{DA}} \leftarrow -\frac{1}{\beta}\sum_{i\in N}\log\left(\sum_{j\in K}e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}\right)$
19: compute $E \leftarrow -\frac{\partial\beta F_{\text{DA}}}{\partial\beta}$
20: decrease, θ
21: **end while**
22: increment β: $\beta+\delta\beta$ *//Annealing*
23: *//Update θ such that tour-length remains constant between each β update*
24: re-initialize θ: $\theta-\frac{\delta\beta}{\beta}\left(\frac{\Delta E}{\Delta L}+\theta\right)$
25: **end while**
26: **return:** $\mathcal{Y}$, $\{p(j|i)\}$, $\Pr(\mathcal{R})$

We provide an overview of the MEP solution algorithm in Algorithm 1. Each iteration of the algorithm is over a single value of the annealing parameter β (lines 7–17). For a constant β, the secondary Lagrange multiplier θ must be updated to ensure that the tour-length remains constant during each β update. In the absence of this update, the results can be less meaningful or far from optimal, as was observed in Vakhutinsky and Golden (1994). We now describe the algorithm for the control of the secondary Lagrange multiplier.

Controlling the secondary Lagrange multiplier

We now present a procedure to determine a new initial value of θ during each β update, such that the free tour-length is kept constant (Rose, 1990). For the sake of brevity, the update procedure is derived only for the basic TSP. The approach is quite general and is easily extended to address any of the variants of the classical TSP. For notational convenience, we use $\theta^*, \mathcal{Y}^*$ and L to denote the optimum (local) value of secondary Lagrange parameter, optimal set of partitions and optimal free tour-length at a given value of β, respectively. Finally, let $F^*_{\text{TSP}} \triangleq F_{\text{TSP}}(\mathcal{Y}^*,\theta^*) = F(\mathcal{Y}^*)$ denote the optimal value of the free-energy function in (15).

From Ravindran, Reklaitis, and Ragsdell (2006), for such constrained optimization, we have:

$$\theta^* = -\frac{\partial F^*_{\text{TSP}}}{\partial L}. \tag{38}$$

Therefore from (38), we have:

$$\frac{\partial \theta^*}{\partial \beta} = -\frac{\partial}{\partial \beta}\left[\frac{\partial F^*_{\text{TSP}}}{\partial L}\right] = -\frac{\partial}{\partial L}\left[\frac{\partial F^*_{\text{TSP}}}{\partial \beta} + \sum_k \frac{\partial F^*_{\text{TSP}}}{\partial \mathbf{y}_k}\frac{\partial \mathbf{y}_k}{\partial \beta}\bigg|_{\mathbf{y}_k=\mathbf{y}^*_k}\right] = -\frac{\partial}{\partial L}\left[\frac{\partial F^*_{\text{TSP}}}{\partial \beta}\right], \tag{39}$$

where the last statement is a consequence of the fact that $\frac{\partial F_{\text{TSP}}}{\partial \mathbf{y}_j} = 0$ for all j at the optimum. Using the function form of F from (15), $\frac{\partial F_{\text{TSP}}}{\partial \beta}$ evaluates to:

$$\frac{\partial F}{\partial \beta} = \frac{1}{\beta^2}\sum_{i=1}^{n}\log\left(\sum_{j=1}^{n} e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}\right) + \frac{1}{\beta}\sum_{i,j=1}^{n}\frac{e^{-\beta d(\mathbf{x}_i,\mathbf{y}_j)}}{\sum_{j'=1}^{n} e^{-\beta d(\mathbf{x}_i,\mathbf{y}_{j'})}} d(\mathbf{x}_i,\mathbf{y}_j), \tag{40}$$

which can be re-written as:

$$\frac{\partial F_{\text{TSP}}}{\partial \beta} = -\frac{F_{\text{TSP}}}{\beta} + \frac{1}{\beta}\underbrace{\left[\theta\sum_{j=1}^{n-1} d(\mathbf{y}_j,\mathbf{y}_{j+1}) + \sum_{i,j=1}^{n} p(j|i)d(\mathbf{x}_i,\mathbf{y}_j)\right]}_{\triangleq E}, \tag{41}$$

where $p(j|i)$ is as defined in Section 4.2. Note that the expression E in (41) is related to the free-energy F by:

$$E = \frac{\partial}{\partial \beta}(\beta F_{\text{TSP}}). \tag{42}$$

Thus from (39), (41) and (42), we have:

$$\frac{\partial \theta^*}{\partial \beta} = -\frac{\partial}{\partial L}\left(\frac{E^* - F^*_{\text{TSP}}}{\beta}\right) = \frac{1}{\beta}\left(-\frac{\partial E^*}{\partial L} + \underbrace{\frac{\partial F^*_{\text{TSP}}}{\partial L}}_{=-\theta^*}\right) \quad \text{[from (38)]} \tag{43}$$

Therefore, from (43), we consider the following first-order approximation for θ update:

$$\theta' \approx \theta^* + \frac{\partial \theta^*}{\partial \beta}\Delta\beta \Rightarrow \theta' = \theta^* + \frac{\Delta\beta}{\beta}\left(-\frac{\Delta E^*}{\Delta L} - \theta^*\right), \tag{44}$$

where $\Delta E^*/\Delta L$ is estimated using the last two iterations in θ (before the moment to update β arrives).

Note that (44) holds true for any TSP variant, however, the formulations of free-energy function F and free tour-length L are different for each variant. For instance, in a classical TSP, the free tour-length is simply the cumulative distance between consecutive partitions, whereas in an mTSP the free tour-length includes the total length of all individual tours.

4.7. Inclusion of soft time-window constraints

Our approach to the Vehicle Routing Problem with Time-Windows (VRPTW) is based on solving routing problems in a higher-dimensional space, where the added dimension corresponds to the time-window constraints. For ease of understanding, we first focus on pure scheduling problems without the routing aspect. Consider n shipments which need to be served (e.g. collected from or deposited to a single depot) within their pre-specified corresponding time-windows $\{[t_{i,\text{start}}, t_{i,\text{end}}] : 1 \le i \le n\}$ by a fleet of m vehicles. The objective is to allocate arrival times for each vehicle $\mathcal{Y} = \{\mathbf{y}_j : 1 \le j \le m\}$ to serve the shipments within specified time-windows such that the maximum number of shipments are served. In this context we choose the set of customer 'locations' $\mathcal{X} = \{i : 1 \le i \le n\}$ as the *mid-point* of the associated time-windows, i.e.,

$$i = \frac{t_{i,\text{start}} + t_{i,\text{end}}}{2}. \tag{45}$$

With the above choice of customer locations, the cumulative distance function $D(\mathcal{Y},\mathcal{V})$ for an instance $(\mathcal{Y},\mathcal{V})$ captures the penalty of the actual service time deviating from the *mid-times*. This function is chosen to maintain continuity of the objective function and can also accommodate other penalty functions, ideally, those that are continuous and differentiable. Thus, minimization of the free-energy function in (13) is commensurate with minimizing the total cost incurred for not serving a shipment within the specified time-window. In this introductory paper, we use the deviation from the mid-times of the time-windows as our penalty function; however, in future work, we will expand this to other continuously differentiable functions that have negligible penalty within the allowed time windows and higher penalty outside the allowed windows.

To adapt this to the VRPTW, we modify the set of customer locations $\mathcal{X} = \{i : 1 \le i \le n\}$ to not only comprise of geographical location coordinates, but also the 'temporal coordinates' weighted using a *speed*-like factor λ, i.e.,

$$i = \begin{bmatrix} x_i^{(1)} \\ x_i^{(2)} \\ \lambda\dfrac{t_{i,\text{start}} + t_{i,\text{end}}}{2} \end{bmatrix}, \tag{46}$$

where $(x_i^{(1)}, x_i^{(2)}) \in \mathbb{R}^2$ represent the geographical locations of the ith customer, and $\lambda > 0$ is an additional parameter that captures the trade-off between the geographical optimization and the scheduling optimization. More specifically, we solve the VRPTW problem by solving the regular mTSP problems where we replace the customer location by (46). When λ is set to zero, the problem reduces to a routing only problem, while setting λ to a very high value essentially disregards the routing aspect and only solves the pure scheduling problem described above. In practice, λ is set to a value comparable to the average speed of the vehicles.

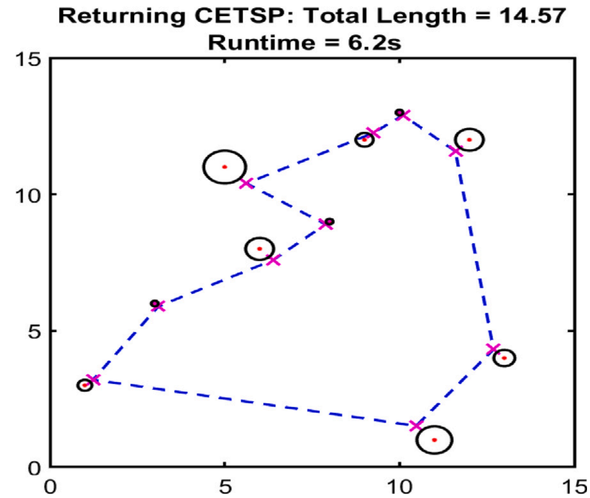


Fig. 6. Illustration of the Close-enough TSP (CETSP) solutions generated using our MEP approach.

5. Computational results and discussion

We now benchmark the proposed MEP heuristics for variants of the routing problem on synthetic and real-world instances. Our algorithms are implemented on an Intel $i7-7700$HQ @ 2.80 GHz machine using MATLAB. The algorithm is benchmarked for both solution quality, as

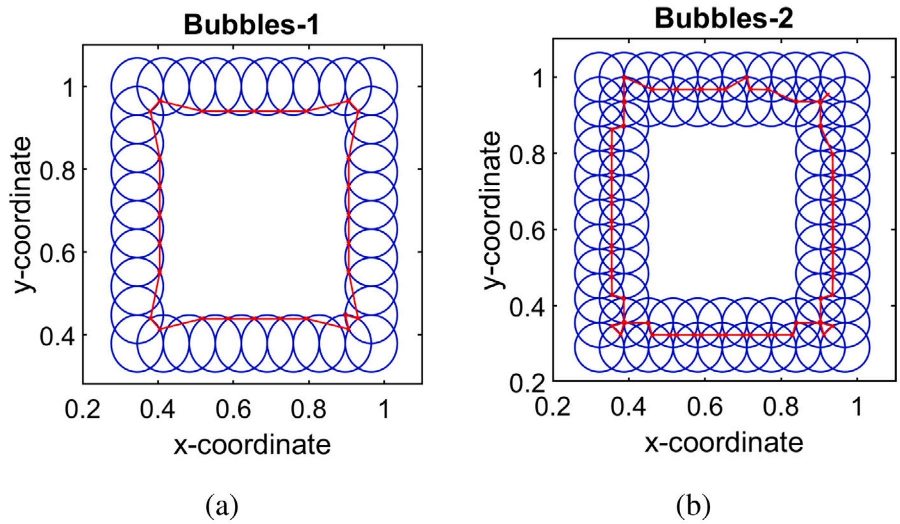
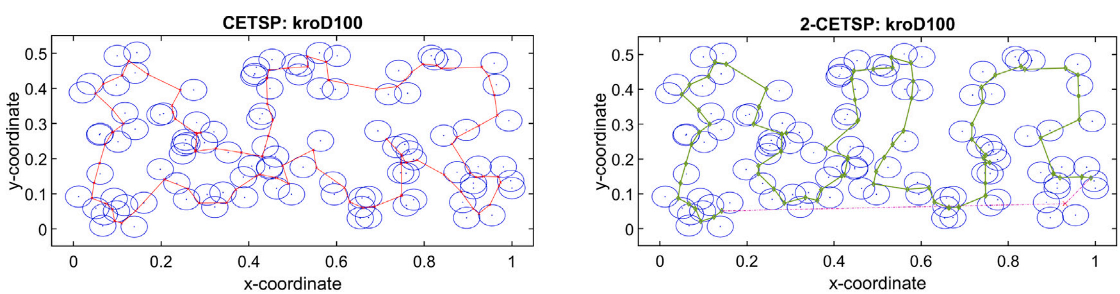


Fig. 7. CETSP solutions on the Bubbles benchmark instances. (a) Bubbles-1: Finds optimal solution, except for the tiny 'hook' at the right-bottom corner. (b) Bubbles-2: MEP uses the intersection of two circles to reduce travel distance.



(a) kroD100: Single vehicle CETSP, total length 0.223

(b) kroD100: 2 vehicle CETSP, total length 0.208. The 2nd vehicle only visits points at the bottom right corner

Fig. 8. Illustration of the MEP approach solutions for CETSP instances, with a single vehicle and multiple vehicles.

well as the associated runtime. Recall that our MEP implementation has not been optimized for computational runtime, for example, by parallelizing the computation of probabilities. Hence, the runtime comparisons are largely conservative. The MEP framework is benchmarked against the most widely used heuristic in the VRP literature, namely the Simulated Annealing (SA) algorithm. For consistency, we have used the publicly available optimized SA implementation from the Yarpiz Project (The Yarpiz Project, 2019).

As before, the computational analysis is carried out in the increasing order of complexity of the various routing problems - (i) close-enough TSP, (ii) capacitated vehicle routing problem, (iii) vehicle routing problem with time-windows. We benchmark the MEP variant of CETSP on TSPLIB (Reinelt, 1991) instances against the state-of-the-art Steiner Zone heuristic for equal radii case proposed by Mennell (2009). Additionally, we illustrate our approach for CETSP on a synthetic dataset with unequal radii which demonstrates the flexibility of our framework. For CVRP, we consider both the real-world instances (where we compare our algorithm against the SA and the cluster-first route-second approaches), as well as the best-known solutions (best of best over all previous approaches reported in Laporte et al. (2014)) of CMT-14 (Christofides, Mingozzi, & Toth, 1979) instances. Finally, we benchmark the MEP approach against the SA algorithm on a real-world instance for VRPTW, and illustrate the key difference in solution quality of both these approaches.

5.1. Benchmarking CETSP

Fig. 6 shows the implementation results for the CETSP on randomly generated dataset with 10 customer locations, with additional radii parameters. For the CETSP, it is difficult to determine optimality of our solution, because this is much more difficult to check manually and unlike the standard TSP, benchmarks and corresponding optimal solutions are not standardized. We compare our MEP heuristic against the larger, 100-cities (customers) data (kroD100 from TSPLIB Reinelt, 1991) tested by Mennell for equal radii of 11.697 (Mennell, 2009). Mennell achieves an average tour length of 58.54 units on instances with a 0.3 overlap ratio on the data (they do not provide details on computation time). Our MEP-based heuristic finds an average optimal tour length of 64.99 units in 863 s. Our tests on Mennell's 48 instances created for the CETSP reveal the following. For instances with low overlap between customers' circles, such as *bubbles1* (Fig. 7) and *concentric1*, we get slightly better solutions than Mennell's solutions; but as the level of overlap increases, Mennell's solutions dominate ours in terms of tour distances. This is because our MEP method aims to visit points on the periphery of the circle around each customer because the interior of the circle is also penalized, whereas other methods such as Mennell's aim towards visiting highly overlapping zones to cover multiple customers' radii (or circles). To overcome this, in future work, we will minimize the penalty structure for visiting a customer within the allowed radius, and maximize it for visiting outside the radius, as discussed in Section 6. On the other hand, Mennell's and other methods are geared towards the radii around each customer being equal; our MEP method allows for heterogeneous radii for different customers. Moreover, while Mennell's method is geared towards a single vehicle, our approach can accommodate multiple vehicles in the CETSP (Fig. 8). However, due to a lack of benchmark instances for the multiple-vehicle case, we do not have a standard for comparison of our solutions.

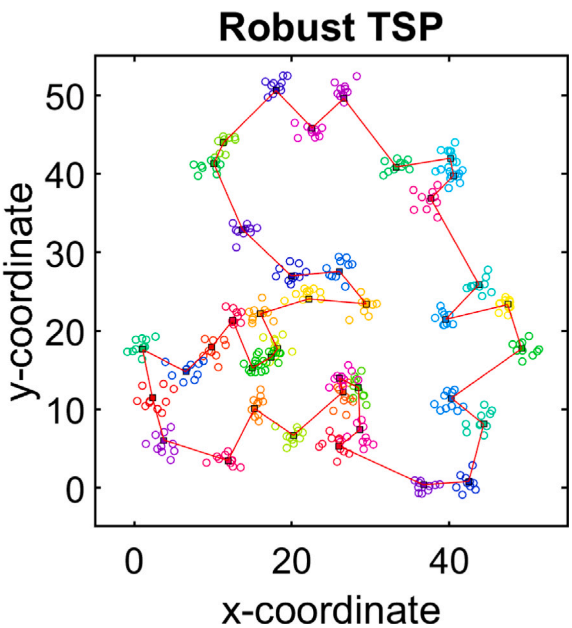


Fig. 9. Robust-TSP: TSL when city location is not known precisely.

Section 3.1 also introduced a robust variant of the basic TSP, also referred to as the robust-TSP and is closely related to the CETSP. In robust-TSP, the customer locations are known only probabilistically, and the objective is to minimize the cumulative expected route length. Fig. 9 shows the results for one such synthetic problem instance in which the locations of the customers are known with uncertainty. The MEP algorithm assigns probabilities to each location to find robust routes that can be executed with the least possible deviations in expectation.

5.2. CVRP

We first report the performance on a real-world dataset of 60 customers in Gurugram city, India, using three distinct approaches. The first approach (Fig. 10a) is based on cluster-first route-second heuristic, where customer locations are first clustered based on the respective capacities, and routes within each cluster are obtained using the proposed MEP framework for routing. This approach, though easily scalable to a large number of customers, produces routes that are largely non-overlapping due to clustered set of customer locations. The second approach is based on SA (Fig. 10b) algorithm. The final approach (Fig. 10c) is based on the direct MEP algorithm described in Algorithm 1. Since the approach directly concurrently incorporates clustering and routing, the routes produced by it are overlapping (and hence the total travel distance is shorter). Recall that if the routes are non-overlapping, then a far-off cluster would be served by one set of vehicles, while a nearby cluster would be served by another set, thereby giving up the opportunity to serve a nearby customer on its way using a vehicle scheduled to serve another customer located significantly further away from the depot. Thus, overlapping routes potentially reduce the number of vehicles and total distance traveled. The direct MEP-based approach results in solutions with much lower costs (by $\sim 32\%$) than the SA algorithm, along with lower computation times. More specifically, the run-time and the total lengths for the three approaches are obtained as - (a) Run-time: 46.6 s, Sq. total length: 789.78 km. (b) Run-time: 51.2 s, Sq. total length: 706.46 km. (c) Run-time: 17.08 s, Sq. total length: 552.44 km.

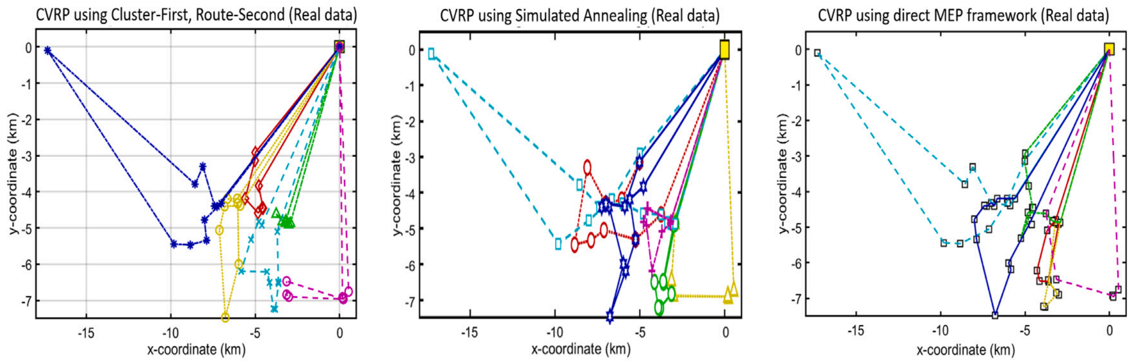
Table 2
Performance on CMT-14 instances.

Dataset	Best known	MEP approach
50C	524.61	537.07
75C	835.26	891.58
100C	826.14	927.06
150C	1028.42	1160.7
199C	1291.29	1476.2
50CD	555.43	556.68
75CD	909.68	929.69
100CD	865.94	900.80
150CD	1162.55	1207.93
199CD	1395.85	1494.96
120C	1042.11	1090.72
100C	819.56	839.68
120CD	1541.14	1598.09
100CD	866.37	900.97

For a more thorough comparison, we draw from the standard CMT-14 instances that have been extensively used as benchmarks in the literature. Table 2 summarizes the performance of the MEP-based cluster-first route-second approach on the CMT-14 benchmark instances (Laporte et al., 2014) for the CVRP. There are 14 instances in the CMT-14 benchmark dataset, with the number of customers ranging from 50–200. Capacity requirements for customers are heterogeneous. Vehicle capacities are also heterogeneous. Additionally, there are constraints on the total time taken by each vehicle to serve its allotted customers. Serving a customer incurs additional time, which is often referred to as the *service time*. Subject to these constraints, we adopt a cluster-first-route-second approach embedded with MEP; that is, we first cluster the customers based on their capacity requirements and geographical coordinates, and find economic routes within each cluster. We further adjust these routes using basic swap and insertion methods to ensure feasibility of the final solution, i.e., customers between two neighboring routes are randomly swapped to generate new routes or a customer is randomly removed from a route and inserted into another route. The process is repeated a few times until an improved solution is obtained or the set maximum allowable iterations are elapsed. Solutions produced by our approach to each of these instances utilize an equal number of vehicles as the number suggested by the best-known solutions over all approaches in the CMT-14 dataset. Moreover, the cumulative distance traveled by the vehicles (salesmen) is within 6.2% of the best-known solutions on average. The average CPU time per instance is approximately 2.5 min.

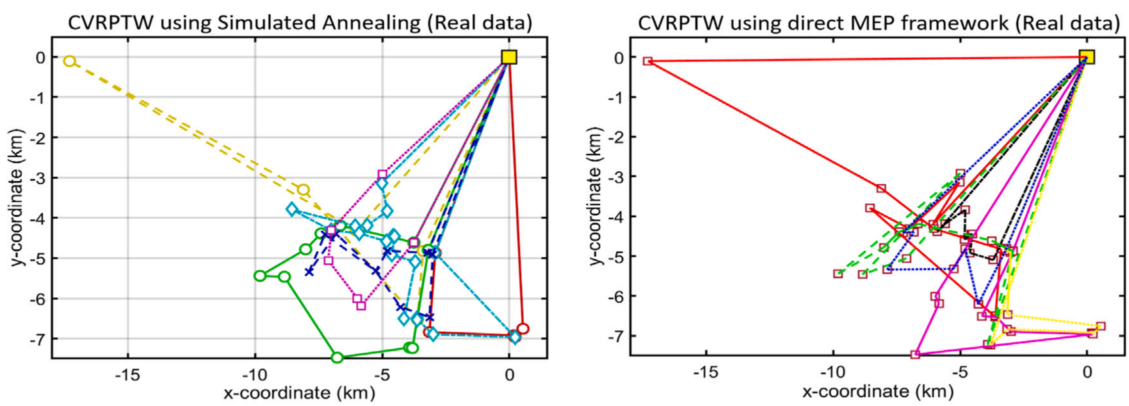
5.3. VRPTW

We again work with the real-world instance using the datasets for Gurugram city, India, with 60 shipments on average, with delivery time-windows of 8:00–11:00, 10:00–13:00 and 16:00–19:00 h respectively as specified by the delivery service provider. Fig. 11 shows the performance of the MEP approach and the SA approach for the Gurugram city data with 60 shipments. Although time-windows are considered soft, the MEP solution satisfies all time-window constraints, with a total distance lower by $\approx 17\%$ than SA. It is slightly slower at 49.5 s versus 44.1 s for the SA. The difference in total travel distance between the two approaches can also be observed qualitatively. In particular, the yellow-route in Fig. 11a, attributed to the point located at the top-left corner, covers only a few customer locations (as opposed to the red route in Fig. 11b). Since this point is located farthest from the depot and the total length of the route covering that point is at least twice the distance of the point from the depot, it is desirable to cover multiple other points on the route in order to minimize total travel distance. The nature of the solution produced by the MEP framework exhibits this behavior.



(a) (b) (c)

Fig. 10. Comparison of the proposed simultaneous clustering and routing approach with the cluster-first route-second method and simulated annealing (SA) for the 60 customers Gurugram data. (a) Cluster-first route-second-Run-time: 46.6 s, Sq. total length: 789.78 km. (b) SA-Run-time: 51.2 s, Sq. total length: 706.46 km. (c) MEP approach-Run-time: 17.08 s, Sq. total length: 552.44 km.



(a) (b)

Fig. 11. Performance of the simulated annealing (SA) and the proposed MEP approach for 60 customers Gurugram data with three service time-windows (a) SA: Run-time 49.5 s, (b) MEP approach: Run-time 44.1 s.

6. Algorithmic analysis and discussion

Flexibility of the algorithm

The MEP approach for routing problems is flexible and generalizable. With distance functions modified appropriately, the MEP based framework is applicable to several variants. Several constraints, including but not limited to, returning/non-returning constraints, capacity constraints, time-window constraints, and close-enough constraints are easily incorporated into the framework and contribute to the flexibility of the MEP approach.

Scalability

The inclusion of the entropy term for making the approach insensitive to initial choice of routes comes at the expense of requiring global computations to be made for evaluating Gibbs distribution and partitions. However as the annealing parameter grows larger (i.e. $\beta \to \infty$), the contributions of the "far-off" customer locations in evaluating Gibbs distribution becomes progressively smaller. In fact, in the limiting case, the contribution of customer locations that are not "nearest neighbors" is zero. This property can be used for significantly reducing run times as explained in (a) and (b) below.

(a) Localizing Gibbs estimates

At each value of the annealing parameter β, the computations of Gibbs distributions are global in the sense that distances of each customer to all the facilities are used. However, as the algorithm progresses with higher values of β, the Gibbs distributions p assume approximately binary values 0 or 1. These approximations become exact as $\beta \to \infty$. This property can be exploited to make the proposed approach scalable. A sufficiently large threshold $\delta > 0$ and all customer locations $\mathbf{x}_l$ that satisfy $d(\mathbf{x}_l, \mathbf{y}_j) > \delta$ can be ignored in the computation of $p(j|l)$. This approximation makes our approach scalable. Since large values of the distances correspond to smaller value of the associated Gibbs distribution, the truncated terms have only small effects on the computation of $p(j|l)$ and the partition location $\mathbf{y}_j$ thereof. Automating the choice of δ is ongoing work.

(b) Keeping secondary Lagrange multiplier fixed

The approach for routing problems proposed in this manuscript requires computing the optimal value of the secondary Lagrange multiplier $\theta(\beta)$ at each value of the annealing parameter β. On the other hand, historically, Elastic Net based approaches to routing problems have largely depended on keeping the secondary multiplier θ fixed for all values of the primary Lagrange multiplier β (Yuille, 1990) or varying θ as $\theta \propto \frac{1}{\sqrt{\beta}}$ (Durbin & Willshaw, 1987). Choosing θ as a deterministic function of β would result in significant reduction in complexity of our routing algorithm, though the reduction in complexity may come at the cost of marginal increase in optimal route-length. A detailed analysis of this tradeoff is part of our ongoing work.

Addressing penalties within allowed radii and time-windows

The current CETSP formulation contains a penalty for visiting a customer, both inside and outside of the allowed radius around each customer; similar to the time-window penalty for VRPTW that imposes some (low) penalty even for visiting within the allowed time-window. However, this should be zero. This can be addressed by setting the derivative of the distance function $d_{\mathrm{CE}}(\mathbf{x}_i, \mathbf{y}_j, \rho_i)$ with respect to $\mathbf{y}_j$ to zero whenever $\mathbf{y}_j$ exists within ρ_i distance from the customer location $\mathbf{x}_i$. This negates the penalty incurred for visiting a customer within the specified radius and should help this heuristic identify more accurate solutions.

7. Conclusions and future work

In this paper, we introduced a unified flexible algorithmic approach, which we refer to as the MEP approach, which is applicable for a large class of routing problems, and demonstrated that this method can be used to formulate and solve multiple classes of routing problems including VRPs and CETSPs and their variants. We show that our MEP approach improves algorithmically upon its precursors, the Deterministic Annealing (DA) approach and the Elastic Net (EN) approach in this category, by devising a principled approach to tuning hyperparameters for the approach. We also discuss that the parameter update step is characterized as a gradient descent step. Our approach exhibits fast run times with significant parallelization and solution quality close to existing techniques. We test the performance of our approach on real-world data and benchmark instances for the VRP and demonstrate that the MEP approach can generate solutions within 6.2% of the best known solutions. We also test our approach on recent benchmarks for the CETSP and discover that our solutions exhibit minor differences to the best known solutions — specifically, our approach performs better when overlap between the circles of customers is small, compared to Mennell's approach. On the other hand, our approach is more flexible in that it can be more easily generalized to multiple vehicles in the CETSP. Our approach's flexibility, as well as ability to achieve good solution quality compared to best of best-known solutions, make it a suitable candidate for incorporating into hybrid methods and metaheuristics that leverage the strengths of multiple techniques. Our work indicates the promise of this method and its potential to be enhanced in combination with other heuristics, and to be competitive with best-known solutions in the literature. We also discussed enhancements being explored in ongoing work, to improve scalability of our algorithm and to modify penalty functions for CETSP to improve accuracy.

The generalizability aspect of the MEP framework is quite appealing; however, similar to most heuristics, the framework lacks optimality guarantees. While we show empirically that the performance of the proposed algorithm is comparable to several benchmark routing algorithms, it is desirable to characterize the worst-case performance bounds analytically, and will be discussed in our future work. Our initial analysis suggests that for a class of combinatorial optimization problems that involve bipartite graph matching, the MEP framework finds the true optimal solution. We are working on extending such suboptimality guarantees for the class of problems discussed in this work, i.e., VRP and its variants.

Another future direction points to improving the modeling and formulation of distances in CETSPs. Currently, our formulation requires that points of visit lie only on the periphery of the circle of uncertainty around a location. However, sometimes if the neighboring circles are highly overlapping, it is often desirable to visit points that are common to both the circles and lie strictly inside each of them. This would result in shorter routes and fewer points of visit. Finally, we aim to extend the current formulation to dynamic real-time instances and problems involving ride-sharing and dynamic pickup and delivery.

CRediT authorship contribution statement

Mayank Baranwal: Methodology, Software, Formal analysis, Investigation, Writing – original draft, Writing – review & editing. **Lavanya Marla:** Conceptualization, Methodology, Data curation, Supervision, Resources, Writing – original draft, Writing – review & editing, Project administration. **Carolyn Beck:** Conceptualization, Methodology, Supervision, Resources, Writing – original draft, Writing – review & editing. **Srinivasa M. Salapaka:** Conceptualization, Methodology, Supervision, Resources, Writing – original draft, Writing – review & editing.

Data availability

The authors do not have permission to share data.

References

Baranwal, M., Parekh, P., Marla, L., Salapaka, S. M., & Beck, C. (2016). Vehicle routing problem with time windows: A deterministic annealing approach. In *2016 American control conference (ACC)* (pp. 790–795). IEEE.

Basuki, M., Hidayat, M., Aji, F., et al. (2019). Application of saving matrix methods and cross entropy for capacitated vehicle routing problem (CVRP) resolving. In *IOP conference series: Materials science and engineering, Vol. 462*. IOP Publishing, Article 012025.

Behdani, B., & Smith, J. C. (2014). An integer-programming-based approach to the close-enough traveling salesman problem. *INFORMS Journal on Computing*, *26*, 415–432.

Bland, R., & Shallcross, D. (1987). *Large traveling salesmen problems arising from experiments in x-ray crystallography: A preliminary report on computation*: *Technical report*, Cornell University Operations Research and Industrial Engineering.

Burke, E. K., Hyde, M., Kendall, G., Ochoa, G., Ozcan, E., & Woodward, J. R. (2010). A classification of hyper-heuristic approaches. In *Handbook of metaheuristics* (pp. 449–468). New York: Springer.

Carrabs, F., Cerrone, C., Cerulli, R., & Gaudioso, M. (2017). A novel discretization scheme for the close enough traveling salesman problem. *Computers & Operations Research*, *78*, 163–171.

Carrabs, F., Cerrone, C., Cerulli, R., & Golden, B. (2020). An adaptive heuristic approach to compute upper and lower bounds for the close-enough traveling salesman problem. *INFORMS Journal on Computing*, *32*(4), 1030–1048.

Chen, P., Huang, H.-K., & Dong, X.-Y. (2010). Iterated variable neighborhood descent algorithm for the capacitated vehicle routing problem. *Expert Systems with Applications*, *37*, 1620–1627.

Christofides, N., Mingozzi, A., & Toth, P. (1979). The vehicle routing problem. In *Combinatorial optimization* (pp. 315–338). Chichester, UK: Wiley.

Clarke, G., & Wright, J. W. (1964). Scthheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, *12*, 568–581.

Cordeau, J.-F., Laporte, G., & Mercier, A. (2001). A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society*, *52*(8), 928–936.

Creput, J.-C., Hajjam, A., Koukam, A., & Kuhn, O. (2012). Self-organizing maps in population based metaheuristic to the dynamic vehicle routing problem. *Journal of Combinatorial Optimization*, *24*, 437–458.

Desaulniers, G., Desrosiers, J., Erdmann, A., Solomon, M. M., & Soumis, F. (2002). VRP with pickup and delivery. *The Vehicle Routing Problem*, *9*, 225–242.

Dong, J., Yang, N., & Chen, M. (2007). Heuristic approaches for a tsp variant: The automatic meter reading shortest tour problem. In *Extending the horizons: Advances in computing, optimization, and decision technologies* (pp. 145–163). Springer.

Dror, M. (1994). Note on the complexity of the shortest path models for column generation in VRPTW. *Operations Research*, *42*(5), 977–978.

Dueck, G. (1993). New optimization heuristics: The great deluge algorithm and the record-to-record travel. *Journal of Computational Physics*, *104*, 86–92.

Dueck, G., & Scheur, T. (1990). Threshold accepting: A general purpose optimization algorithm appearing superior to simulated annealing. *Journal of Computational Physics*, *90*, 161–175.

Durbin, R., Szeliski, R., & Yuille, A. L. (1989). An analysis of the elastic net approach to the traveling salesman problem. *Neural Computation*, *1*(3), 348–358.

Durbin, R., & Willshaw, D. (1987). An analogue approach to the travelling salesman problem using an elastic net method. *Nature*, *326*(6114), 689.

Foster, B. A., & Ryan, D. M. (1976). An integer programming approach to the vehicle scheduling problem. *Operational Research Quarterly*, *27*, 367–384.

Gillett, B. E., & Miller, L. R. (1974). A heuristic algorithm for the vehicle-dispatch problem. *Operations Research*, *22*(2), 340–349.

Gulczynski, D., Heath, J., & Price, C. (2006). The close enough traveling salesman problem: A discussion of several heuristics. In *Perspectives in operations research: Papers in honor of saul gass' 80th birthday* (pp. 271–283). Springer.

Helsgaun, K. (2000). An effective implementation of the lin-kernighan traveling salesman heuristic. *European Journal of Operational Research*, *126*, 106–130.

Hiquebran, D., Alfa, A., Shapiro, J., & Gittoes, D. (1993). A revised simulated annealing and cluster-first route-second algorithm applied to the vehicle routing problem. *Engineering Optimization*, *22*(2), 77–107.

Jaynes, E. T. (1957). Information theory and statistical mechanics. *Physical Review*, *106*(4), 620.

Kale, N. V., & Salapaka, S. M. (2012). Maximum entropy principle-based algorithm for simultaneous resource location and multihop routing in multiagent networks. *IEEE Transactions on Mobile Computing*, *11*(4), 591–602.

Kytojoki, J., Nuorito, T., Nuorito, T., Braysy, O., & Gendreau, M. (2007). An efficient variable neighborhood search heuristic for very large scale vehicle routing problems. *Computers & Operations Research*, *34*, 2743–2757.

Labadie, N., Prins, C., & Prodhon, C. (2016). *Metaheuristics for vehicle routing problems* (1st ed.). Wiley-IEEE Press.

Laporte, G., Ropke, S., & Vidal, T. (2014). Chapter 4: Heuristics for the vehicle routing problem. In *Vehicle routing: Problems, methods, and applications* (2nd ed.). (pp. 87–116). SIAM.

Li, F., Golden, B. L., & Wasil, E. A. (2005). Very large-scale vehicle routing: new test problems, algorithms, and results. *Computers & Operations Research*, *32*, 1165–1179.

Lin, S., & Kernighan, B. W. (1973). An effective heuristic algorithm for the traveling salesman problem. *Operations Research*, *21*, 498–516.

Lloyd, S. P. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, *28*(2), 129–137.

Mastan, S., Balakrishnan, U., & Sankar Sekhar Raju, G. (2019). *A quick heuristic algorithm for travelling salesman problem*: *Technical report*, SSRN Electronic Repository.

Matai, R., Singh, S. P., & Mittal, M. L. (2010). Traveling salesman problem: an overview of applications, formulations, and solution approaches. In *Traveling salesman problem, theory and applications, Vol. 1*. Croatia: InTech.

Mennell, W. K. (2009). *Heuristics for solving three routing problems: Close-enough traveling salesman problem, close-enough vehicle routing problem, sequence-dependent team orienteering problem* (Ph.D. thesis), University of Maryland.

Miranda-Bront, J. J., Curcio, B., Méndez-Díaz, I., Montero, A., Pousa, F., & Zabala, P. (2017). A cluster-first route-second approach for the swap body vehicle routing problem. *Annals of Operations Research*, *253*(2), 935–956.

Nagata, Y., & Braysy, O. (2009). Edge assembly-based memetic algorithm for the capacitated vehicle routing problem. *Networks*, *54*, 205–215.

Nelson, M. D., Nygard, K. E., Griffin, J. H., & Shreve, A. E. (1985). Implementation techniques for the vehicle routing problem. *Computers & Operations Research*, *12*, 273–283.

Osman, I. H. (1993). Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Annals of Operations Research*, *41*, 421–451.

Paessens, H. (1988). The savings algorithm for the vehicle routing problem. *European Journal of Operational Research*, *34*, 336–344.

Parekh, P. M., Katselis, D., Beck, C. L., & Salapaka, S. M. (2015). Deterministic annealing for clustering: Tutorial and computational aspects. In *American control conference (ACC), 2015* (pp. 2906–2911). IEEE.

Pisinger, D., & Ropke, S. (2007). A general heuristic for vehicle routing problems. *Computers & Operations Research*, *34*.

Plante, R. D., Lowe, T. J., & Chandrasekaran, R. (1987). The product matrix traveling salesman problem: an application and solution heuristic. *Operations Research*, *35*(5), 772–783.

Poggi, M., & Uchoa, E. (2014). Chapter 3: New exact algorithms for the capacitated vehicle routing problem. In *Vehicle routing: Problems, methods, and applications* (2nd ed.). (pp. 59–86). SIAM.

Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research*, *31*, 1985–2002.

Ratliff, H. D., & Rosenthal, A. S. (1983). Order-picking in a rectangular warehouse: a solvable case of the traveling salesman problem. *Operations Research*, *31*(3), 507–521.

Ravindran, A., Reklaitis, G. V., & Ragsdell, K. M. (2006). *Engineering optimization: Methods and applications*. John Wiley & Sons.

Reichl, L. E. (1998). *A modern course in statistical physics* (2nd ed.). Wiley, John & Sons Inc..

Reimann, M., Doerner, K. F., & Hartl, R. F. (2004). D-Ants: Savings based ants divide and conquer the vehicle routing problem. *Computers & Operations Research*, *31*, 563–591.

Reinelt, G. (1991). TSPLIB - a traveling salesman problem library. *INFORMS Journal on Computing*, *3*(4), 376–384.

Renaud, J., Boctor, F. F., & Laporte, G. (1996). An improved petal heuristic for the vehicle routing problem. *Journal of the Operational Research Society*, *47*, 329–336.

Ropke, S., & Pisinger, D. (2006). A unified heuristic for a large class of vehicle routing problems with backhauls. *European Journal of Operational Research*, *2004*, 750–775.

Rose, K. (1990). *Deterministic annealing, clustering, and optimization* (Ph.D. thesis), California Institute of Technology.

Rose, K. (1998). Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. *Proceedings of the IEEE*, *86*(11), 2210–2239.

Ryan, J. L., Bailey, T. G., Moore, J. T., & Carlton, W. B. (1998). Reactive tabu search in unmanned aerial reconnaissance simulations. In *Proceedings of the 30th winter simulation conference* (pp. 873–880). IEEE Computer Society Press.

Ryan, D. M., Hjorring, C., & Glover, F. (1993). Extensions of the petal method for vehicle routing. *Journal of the Operational Research Society*, *44*, 289–296.

Salapaka, S., Khalak, A., & Dahleh, M. (2003). Constraints on locational optimization problems. In *Proceedings of the 42nd IEEE conference on decision and control, Vol. 2* (pp. 1741–1746). IEEE.

Shaw, P. (1997). A new local search algorithm providing high quality solutions to vehicle routing problems.

Shuttleworth, R., Golden, B. L., Smith, S., & Wasil, E. (2008). Advances in meter reading: Heuristic solution of the close enough traveling salesman problem over a street network. In *The vehicle routing problem: Latest advances and new challenges* (pp. 487–501). Springer.

Stodola, P., Michenka, K., Nohel, J., & Rybanskỳ, M. (2020). Hybrid algorithm based on ant colony optimization and simulated annealing applied to the dynamic traveling salesman problem. *Entropy*, *22*(8), 884.

Stone, J. (1992). The optimal elastic net: Finding solutions to the travelling salesman problem. In *Artificial neural networks* (pp. 1077–1080). Elsevier.

Subramanian, A., Uchoa, E., & Ochi, L. S. (2013). A hybrid algorithm for a class of vehicle routing problems. *Computers & Operations Research*, *40*, 2519–2531.

Tarantilis, C. D., Anagnostopoulou, A. K., & Repoussis, P. P. (2013). Adaptive path relinking for vehicle routing and scheduling problems with product returns. *Transportation Science*, *47*, 356–379.

The Yarpiz Project (2019). Simulated annealing in MATLAB. http://yarpiz.com/223/ypea105-simulated-annealing/.

Thompson, P. M., & Psaraftis, H. N. (1993). Cyclic transfer algorithms for multivehicle routing and scheduling problems. *Operations Research*, *41*, 935–946.

Toth, P., & Vigo, D. (2002). *The vehicle routing problem*. SIAM Publications.

Toth, P., & Vigo, D. (2014). *Vehicle routing: Problems, methods, and applications* (2nd ed.). SIAM.

Vakhutinsky, A. I., & Golden, B. L. (1994). Solving vehicle routing problems using elastic nets. In *Proceedings of the 1994 IEEE international conference on neural networks, ICNN, Vol. 7* (pp. 4535–4540). IEEE.

Vidal, T., Crainic, T. G., Genderau, M., & Prins, C. (2013). Heuristics for multi-attribute vehicle routing problems: A survey and synthesis. *European Journal of Operational Research*, *231*, 1–21.

Vidal, T., Crainic, T. G., Genderau, M., & Prins, C. (2014). A unified solution framework for multi-attribute vehicle routing problems. *European Journal of Operational Research*, *234*, 658–673.

Vidal, T., Craininc, T. G., Gendreau, M., Lahrichi, N., & Rei, A. (2012). A hybrid genetic algorithm for multidepot and periodic vehicle routing problems. *Operations Research*, *60*, 611–624.

Xu, Y., Salapaka, S. M., & Beck, C. L. (2014). Clustering and coverage control for systems with acceleration-driven dynamics. *IEEE Transactions on Automatic Control*, *59*(5), 1342–1347.

Yang, K., You, X., Liu, S., & Pan, H. (2020). A novel ant colony optimization based on game for traveling salesman problem. *Applied Intelligence: The International Journal of Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies*, *50*(12), 4529–4542.

Yuan, B., Orlowska, M., & Sadiz, S. (2007). On the optimal robot routing problem in wireless sensor networks. *IEEE Transactions on Knowledge and Data Engineering*, *19*, 1251–1261.

Yuille, A. L. (1990). Generalized deformable models, statistical physics, and matching problems. *Neural Computation*, *2*(1), 1–24.

Zachariadis, E. E., & Kiranoudis, C. T. (2010). A strategy for reducing the computational complexity of local search-based methods for the vehicle routing problem. *Computers & Operations Research*, *37*, 2089–2105.

Zhang, W., Yang, D., Zhang, G., & Gen, M. (2020). Hybrid multiobjective evolutionary algorithm with fast sampling strategy-based global search and route sequence difference-based local search for VRPTW. *Expert Systems with Applications*, *145*, Article 113151.