© 2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Latency-Aware Optimization of Submarine Communication Cable Systems with Trunk-and-Branch Topologies

Tianjiao Wang, Zengfu Wang, Bill Moran, Xinyu Wang, Chao Guo, Moshe Zukerman, Life Fellow, IEEE

Abstract—This paper provides an optimized cable path planning solution, called Lagrangian Fast Marching (LAFM), for the most popular submarine cable system topology, i.e., trunk-and-branch, on the undersea surface of the earth modeled by a triangulated 2D manifold in a 3D Euclidean space. We formulate the problem of a cable system design optimization as a Steiner minimal tree problem, where each Steiner node models a branching unit (BU).

Our objective is to minimize the total cost of the cable system; this total cost is defined to include both actual cable and BU costs — including laying as well as a measure of risk associated with the location and topography. We minimize this total cost while imposing latency constraints limiting the length of cable between specified pairs of nodes. As most BUs in practice are Y-shaped, each Steiner node is assumed to have three branches, in accord with the theory of Steiner trees, at least in the Euclidean plane.

This paper discusses both methodological ideas related to the general problem as well as providing two algorithms, LAFM-I and LAFM-II, to solve the constrained optimization problem. We have proved that LAFM-I can find the optimal solution for cable systems with one latency constraint. We also show that LAFM-II provides a solution with provable bounds for problems with multiple latency constraints. We also find optimal solutions (zero gap between the bounds) for examples with two and with four latency constraints. In addition, we demonstrate the superiority of our LAFM method over an algorithm based on simulated annealing (SA), and demonstrate the applicability of LAFM-I, LAFM-II and SA to realistic scenarios based on real-world data.

Index Terms—Communication cable systems, optimization, latency constraints, Lagrangian, Steiner minimal tree, branching units, submarine manifold

I. INTRODUCTION

Over the last several decades, there has been an explosive growth of internet traffic. A recent report by Cisco indicates that nearly two-thirds of the global population will have internet access by 2023 [1]. At that time, there will be 5.3 billion total internet users, up from 3.9 billion in 2018. In the study of Anja et al. [2], the traffic volume changes follow demand changes, causing a “moderate” traffic surge of 15-20% during lockdown for the Internet Service Provider/Internet Exchange Points. We anticipate that post-COVID-19, living habits will change significantly, with the current increased popularity of working and learning from home. Ever-increasing internet traffic brings greater burdens on global network transmission and, in particular, for the submarine cable network. More cables and networks are needed to support the traffic. In this context, it is of great importance to have efficient and reliable solutions for submarine cable path planning.

One important consideration of cable system design is the total construction cost of the submarine cable system, which is significant. For example, the total construction cost of the EllaLink cable system (with a total cable length of 9,300 km) is $200,000,000 and that of the Pakistan-East Africa Connecting Europe cable system (with a total cable length of 15,000 km) is $425,000,000 [1]. Accordingly, minimizing cable cost (subject to the considerations mentioned above) is essential in constructing a submarine cable system [3–5]. The cost of a cable system includes the costs of the cables, cable laying, and branching units (BUs). The cable laying cost may include labor, materials and permits, etc. [4].

The cost of submarine cable construction is currently estimated at $24,000-$25,000 per kilometer, resulting in high prices for large-scale submarine cable systems. Accordingly, the length of the cable is an important performance measure that we aim to minimize. Another important performance measure is the depth of the earth’s surface where the cable is laid. The International Cable Protection Committee (ICPC) [6] strongly recommends laying the cable in safer deep waters to reduce various risks caused by human-related activities such as fishing. In this paper, in our optimization cost function, we therefore impose lower costs in deeper waters. In this way, our optimization uses location-dependent data, i.e., earth surface depth, as in [7]. Laying cables on the seabed has several advantages over land. First, the cost of laying on the seabed is normally lower and the construction speed is faster than on land. Notice that cables laid on land need to be underground or above ground on pylons, and may go through private property. Secondly, seawater can prevent external optical and magnetic waves and interference, thereby improving the communication quality of the optical cable. Thirdly, compared with cables on...
land, submarine cables are less susceptible to interference and damage by human activities [8]. Accordingly, in our objective weighted cost function the cost on land is higher than undersea and the cost is decreased with depth because, as mentioned in [6], cables are safer in deeper waters.

In practice, the industry generally uses a manual approach, based on expert experience, for path planning of submarine cables. Currently, MakaiPlan [9], an industry-leading software for submarine cable route design is available on the market. It provides some tools necessary for cable route planning and engineering. MakaiPlan only provides path planning of submarine cables connecting two nodes, but does not provide any functions concerning extensions of the submarine cable network. The length of long-haul cables may be thousands of kilometers, so manual path planning without a scalable, automated software tool is costly and may not achieve the desired optimal cost-risk trade-off. Automatic cable path planning, for submarine cables, involving optimization of objectives associated with cost and risk while also meeting quality of service requirements, can enhance the commonly used manual approach by cable path and system planners, designers and surveyors. Industry cable planners, designers and surveyors cannot and will not use the automatic method and solutions proposed in this paper directly to design cable routes. Nevertheless, consideration of the automatic solutions informs the manual approach, especially, where there are latency requirements between specific pairs of nodes. For example, the automatic-algorithm solutions may indicate areas where more seabed data needs to be collected. On the other hand, cost functions can be adjusted based on practical feedback from industry cable planners, designers and surveyors, leading to more accurate solutions.

In addition to the depth of the earth’s surface, cable path planning must take account of a wide range of other location-dependent issues, including earthquake-prone areas, surfaces that have steep slopes or harmful rocks, heavy fishing activity areas, and areas that involve specific legal requirements, as discussed in [4, 5, 10–12]. Path planners may use longer cables to avoid such places. The combination of the considerations of latency-aware design together with the location-dependent cost function is an important attribute of this paper. However, in this paper, for simplicity of exposition, we just take account of the depth of the location-dependent cost function. Nevertheless, our solutions can be extended to consider all these other issues by imposing a summary cost function where the weights of the various design considerations are empirically obtained, using the method presented in [5], if the relevant data is available.

A BU is a device used in submarine communications cable systems to split a cable into multiple cables that can reach multiple destinations. The cost of the BU is determined by its complexity and its number of ports [13]. The cost of a single BU was recently estimated at $1–$2 million [14].

Another critical factor in the design of a cable system is latency; that is, the time required for data to be transmitted between two terminals of communication. The development of cloud computing technology benefits the emerging smart devices with ultra-low latency requirements, i.e., the Internet of Things (IoT) and the 5G applications. However, the high latency caused by long-distance transmission to the conventional centralized clouds through core networks in traditional cloud computing may significantly downgrade the QoE in terms of the response time of latency-sensitive tasks [15]. For transoceanic network services, the routing distances of submarine optical cables can reach thousands of kilometers, accounting for most of the end-to-end delay. Since the refractive index of the optical fiber results in a delay of approximately 5 ms every 1,000 km [16], the delay on the submarine cable increase rapidly with distance, prejudicing low-latency task execution.

For instance, many Industrial Internet-of-Things (IoT) applications, such as motion control systems require low latency (sub-millisecond to few milliseconds) [17]. Latency in image rendering of more than 15 ms in virtual reality (VR) applications can cause motion sickness [18]; a latency of more than 100 ms in online gaming can seriously affect the experience of the gamer; every 100 ms of latency costs Amazon 1% in e-commerce sales [19].

In many cases, traffic between end nodes needs to pass through other nodes. The intermediate nodes further increase total latency because the data packet has to be passed to the upper electrical layer for packet analysis and processing. The latency of intermediate nodes (BUs or CLSs) is non-deterministic and needs to be analyzed on a device-by-device basis. This is related to the way they process incoming signals.

As a promising next-generation BU, the reconfigurable optical add-drop multiplexer (ROADM) BU [20] realizes optical bypass through contained component wavelength selective switching (WSS), which provides signal switching in the pure optical domain and so avoids switching and queueing delays caused by optical/electrical/optical (OEO) conversion and electrical buffering [21, 22]. This paper is based on the assumption that bypass is implemented in every intermediate node [23, 24], so we do not have to consider the intermediate processing delay but mainly focus on the transmission latency, which immensely simplifies our study. In addition, compared with satellites, which is another way for international data transmission, the communications broadcast by submarine cables are not easily affected by weather conditions such as rain or typhoons. Submarine cables have wider bandwidth and are cheaper than satellites. In most cases, lower latency is achieved by submarine cables compared to satellites, except for some cases where Low Earth Orbit satellites may achieve lower latency than some long-haul cables [25, 26].

Submarine cable systems can be designed in various topologies, e.g., point-to-point, ring, mesh, trunk-and-branch [7, 27, 28]. Point-to-point and ring cable systems can be planned using known methods [11, 29]. This paper provides a new method for latency-aware design for trunk-and-branch topology cable systems. Note that our method may not be applicable to some mesh topologies. For cable path planning between two terminals, point-to-point topology is the most popular choice. For more than two terminals, trunk-and-branch topology is the most popular [27]. For example, according to a report released in 2021 [27], among the 356 in-service submarine cable systems, 152 of them contain more than two terminals and 92 of these 152 cable systems use a trunk-and-branch topology. Moreover, among the 59 cable systems that are...
now in the planning stage, 53 of them involve more than two terminals and 34 of them will use a trunk-and-branch topology [27]. Although these tree topology cable systems can cooperate with each other or share some cable branches, when a new cable system is designed, we connect the specified terminals with a tree system. And the latency constraints for some pairs of nodes are required to be satisfied within the new cable system without taking account of communication through other cables. We note that many cable systems that seem to have complex topology such as 2Africa [30], Africa 1 [31], also fall into the category of trunk-and-branch topology and involve no rings.

The problem of cable system design with trunk-and-branch topology can be regarded as a Steiner Minimal Tree (SMT) problem on the submarine surface of the earth. The latter is, in our formulation, represented by a triangulated 2D manifold in a 3D Euclidean space [7, 32]. The well-known SMT problem is described as follows. Given a map of a particular area and the set of terminal nodes \( N \) in this area, design a connected system to connect all the terminal nodes with the objective of minimizing the total length of all the links. In this way, any two terminal nodes are connected, either directly, or through intermediate nodes. Note that the intermediate nodes can either be existing terminal nodes, or extra introduced nodes named Steiner nodes. It is straightforward to show that such an optimal system connecting terminal and Steiner nodes is always a tree — an SMT. The widely studied SMT problem has been proven to be NP-hard [33, 34]. Hence, the solution to the problem is not scalable to large systems. As pointed out in [35], existing SMT algorithms can successfully find the Steiner points based on the angle condition for Steiner points in the plane. But these algorithms cannot be directly applied to the situation of an irregular 2D manifold in 3D Euclidean space.

In this paper, for the first time, we optimize the location of BUs as well as the total system cost, including both the layering cost and BU cost, subject to a latency constraint between specified pairs of terminal nodes. An important part of our methodology is based on solving a constrained SMT problem, in which the BUs are the Steiner nodes, where we optimize the Steiner tree topology as well as the locations of Steiner nodes. The objective function that we aim to minimize is the total cost of the tree system, including the cost of construction of cables and BUs. The contributions of this paper as follows.

- We propose a new method for the problem of latency-aware optimization of a cable system. We treat the problem of minimizing the total construction cost of a cable system with trunk-and-branch topology as an SMT problem. We also take into account latency constraint requirements between specified pairs of nodes. To the best of our knowledge, this is the first attempt to solve this problem in the context of cable path planning.

- We provide, for the first time, a new method that we call Lagrangian Fast Marching (LAFM) for the SMT problem over an irregular 2D manifold in a 3D space with constraints on the total length of each of the cables that connect a pair of nodes of a given set of pairs of nodes.

Based on the LAFM method, we propose the LAFM-I algorithm for single constraint problems and the LAFM-II algorithm for multiple constraint problems.

- We prove that the LAFM-I algorithm provides the optimal solution for a single constraint problem (for a given data resolution) and show that LAFM-II provides a good quality feasible solution with provable bounds for problems involving multiple latency constraints. In fact, in all the realistic examples in this paper, LAFM-II achieves optimal solutions with zero gap between the bounds.

- To provide an alternative approach, we apply Simulated Annealing (SA) [36–40] and compare it with our LAFM method. We apply LAFM-I or LAFM-II and the SA algorithms in several realistic scenarios. In these scenarios the LAFM results (including LAFM-II results) were optimal, so they can serve as a benchmark for the SA results: these were never more than 5% worse than those achieved by LAFM. Note also that SA has further potential benefits in cases where the gap between the LAFM-II bounds is large. Then, combining LAFM-II and SA, by using the result of LAFM-II, the results of both SA and LAFM-II, may be further improved.

The remainder of the paper is organized as follows. In Section II, we survey the related work on submarine cable path planning and SMT problem. The problem and the model of the surface of the earth are introduced in Section III. In Section IV, we present our proposed LAFM method for solving the constrained SMT problem. We then apply our proposed algorithms and SA to realistic scenarios and present the corresponding results in Section V. We provide conclusions in Section IV.

II. RELATED WORK

In this section, we survey the related work in two areas: work on submarine cable path planning, and work on the SMT problem.

A. Submarine cable path planning

Existing work on path planning includes point-to-point [4, 10, 41–46], point-to-area [4, 47–49], and network optimization [7, 28, 32, 50–52].

Point-to-point submarine path planning commands much interest, and many publications focus on it. For example, Wu et al. [10] designed a rectangular topology for laying undersea cables to connect two sufficiently close destinations while maintaining survivability under major disasters. Cao et al. [41, 42] considered the cable path planning problem as a multi-objective optimization problem that includes consideration of cable cost and cable breakage probability. These results on resilient path planning are limited to cables that lie on a plane.

Zhao et al. [46] provided a raster-based path analysis that minimizes the accumulative path cost. They considered both cable lengths and earthquake risk factors, with a model combining a three-dimensional landform map and an earthquake likelihood map. They used the Dijkstra’s algorithm for cable path planning with a cost function including costs associated with cable length and earthquake survivability. A limitation of
their method is that it restricts the geometry of links (in the
grid) that the path can take when it moves from one point in
the grid to another to either a lateral or diagonal link.

The work by Wang et al. [4, 11, 43–45] represents the
earth’s surface by triangulated manifolds and takes account
of cable laying costs and survivability of the cable. Solving
the multi-objective optimization problem using the weighted
sum method yielded a set of path planning solutions that are
Pareto optimal. The weighted sum approach resulted in an
Eikonal equation that was solved by the Fast Marching Method
(FMM). A cost-benefit of up to 17.5% of FMM over Dijkstra
in cable path planning for realistic examples was demonstrated
in [11] and [4].

Designing solutions for connecting a point to an existing
cable network [4, 47–49] and designing a network connecting
multiple points are also crucial in submarine cable path
planning [7, 28, 50]. In much of the existing work on network
planning, a network is modeled by a graph with nodes and
links in 2D space, where the links are all straight lines [53–56].
Guo et al. [55] proposed a heuristic algorithm for enhancing
virtual network design. Their proposed method is resource-
efficient, especially in terms of request acceptance rate and
embedding. Their model is based on an undirected graph,
with straight lines for the connections between terminal pairs.
The model in [56] also involves an undirected graph; there
Ouveysi et al. studied network planning and design under
unstable conditions of link failure and traffic overload. As we
have sought to optimize realistically curved cables over 2D
manifolds in 3D space, we have focused on the application
of algorithms that use FMM for cable path planning, and this
work is most relevant to the present paper.

Wang et al. [4] considered a cable path connecting a
landing station with an arbitrary location on an existing cable.
Msongaleli et al. [50] provided a method for path selection
from a set of submarine cables based on Integer Linear
Programming (ILP). Oostenbrink et al. [47] discussed the
disaster-aware network augmentation problem that aims to
find a new cable connection that minimizes the function of
disaster impact on the cable network as a whole, as well
as on the individual cable segments. They used the great
circle distance for a sphere to compute cable costs. Tran and
Saito [48, 49] proposed an interesting heuristic for optimizing
a weighted set of end-to-end disconnection probabilities under
cable length constraints by either recomputing the routes of
existing links [48] or augmenting the network by adding new
links [49] based on actual seismic hazard data. Their goal was
to find the best geographic route that minimizes the sum of the
end-to-end disconnection probabilities of network links under
cost constraints.

Wang et al. [7] studied a cable network with trunk-and-
branch topology and transformed the path planning problem
into an SMT problem. A dynamic programming method was
proposed to find the location of Steiner points on a triangula-
tion irregular 2D manifold. In their approach, FMM was
used to find the optimal path between pairs of nodes, and a
branch and bound (B&B) method was adopted for topology
enumeration and exclusion.

In our previous work [28], a method named FMM/ILP was
proposed for cable system optimization with the spanning-
tree topology on a triangulated irregular 2D manifold. We
gave consideration to latency constraints without considering
the cost and location optimization of BUs. As optimizing
the location of BUs was not addressed in [28], the work there only
considered the minimum spanning tree problem rather than the
SMT problem. The work in [7] was extended in [32] to include
BU costs, but latency constraints were not considered there. In
a real-world problem, if there are latency constraints, use of
the algorithm developed in [7], will not meet those constraints.

B. SMT problem

For the SMT problem, given a set of N terminals, N ≥ 3,
it is known that the number of full Steiner topologies (FST)
with N – 2 Steiner nodes increases super-exponentially with
N [57, 58].

Many authors have contributed research on the SMT prob-
lem and provided a range of solutions [35, 59–62]. Gilbert and
Pollak [59] proposed a method to solve the SMT problem in
dimensional Euclidean space. Their algorithm enumerates
all Steiner topologies and calculates the relative minimal tree
(RMT) corresponding to each topology. As the number of
Steiner topologies grows super-exponentially with the number
of terminals, their algorithm is computationally intensive.

Another algorithm proposed by Dreyfus et al. [60], based on
dynamic programming, enumerates all possible Steiner tree
topologies to obtain a minimum total length solution. These
two algorithms both have problems with high computational
complexity and can only be applied in a problem with a small
number of terminal nodes. The GeoSteiner algorithm proposed
by Warme et al. [61, 62] is an efficient algorithm to find an
exact solution for the SMT problem. It is based on dividing
the 2D plane that includes the N terminals into several
subsets that are small enough to find an optimal solution for
each of them and then uses a combiner to build a complete
solution. Smith [35] proposed another heuristic SMT algorithm
based on enumeration that generates FST and reduces the
computational complexity by using the B&B method. All of
these algorithms are applicable only to Euclidean spaces. As
mentioned in the Introduction, we seek an SMT algorithm in
an irregular 2D manifold in 3D Euclidean space. Therefore,
the above algorithms are not applicable to our problem.

The dynamic programming algorithm proposed in [7, 32],
which is used to solve the SMT problem in an irregular
2D manifold (in 3D Euclidean space), is the closest to our
algorithm. However, they did not consider latency constraints
between specified pairs of nodes.

C. Summary

Although many publications have been dedicated to subma-
rine cable path planning, none have considered an SMT with
constraints problems. While some studies on SMTs have been
conducted, they do not translate to non-Euclidean manifolds.
We believe that our work is the first to consider the SMT
problem with constraints on a non-Euclidean manifold.
III. PROBLEM MODELING AND FORMULATION

In this section, we first describe our models of submarine landforms and cable laying costs. Then we formulate the cable system optimization problem as an SMT with constraints problem.

A. Problem modeling

We denote a region on the earth’s surface by \( \mathbb{D} \) and \( x_1, x_2, \ldots, x_N \) to be the terminal nodes in \( \mathbb{D} \). Our goal is to optimize the submarine cable system with trunk-and-branch tree topology for these terminal nodes to minimize the overall cost and, at the same time, impose constraints on the distances between specified pairs of terminal nodes. The overall cost of a cable system includes the sum of the costs for all the cables, as well as costs for some newly introduced nodes, named Steiner nodes (represent BUs in cable systems). We reuse notations from [7, 28]: \( \gamma(A, B) \in \mathbb{D} \) (geodesic) is used to denote a cable connecting two end nodes \( A, B \in \mathbb{D} \). Curves in \( \mathbb{D} \) are assumed parametrized according to the natural parametrization, i.e., parametrizing a curve \( \gamma \) as a function of arc length denoted by \( s \), so that a curve \( \gamma : [0, l(\gamma)] \rightarrow \mathbb{D} \) is a function from the interval \([0, l(\gamma)]\), taking values in \( \mathbb{D} \), where \( l(\gamma) \) is the length of the curve \( \gamma \).

To represent the earth’s surface, we use a triangulated piecewise-linear 2D manifold \( M \) in \( \mathbb{R}^3 \) to approximate the region \( \mathbb{D} \). Each node \( h \) in this manifold \( M \) is represented by 3D coordinates \((x, y, z)\), where \( z = \xi(x, y) \) is the altitude of the geographic location \((x, y)\). The more nodes we have, the higher the resolution of the representation, and the more realistic the optima result we obtain. Our aim is to find the optimal solution to our problem given the representation.

The cost per unit length used in our optimization at location \( h \) is \( c(h) \). For details of this representation, the reader is referred to [43]. The total cost used in the optimization of the cable \( \gamma \) is \( C(\gamma) \), which is the objective function cost that we aim to minimize. This is different from the actual cable cost as it is adjusted to include a risk component. Applying the additivity assumption of the cost as discussed in [4, 43], we can write \( C(\gamma) \) as

\[
C(\gamma) = \int_0^{l(\gamma)} c(\gamma(s)) ds.
\] (1)

In most cases, a BU is a Y-shaped cable connector connecting three terminals [13]. In this paper, we only consider BUs with three branches. If two Y-shaped BUs are directly connected to each other, each one of them is connected to two other terminals, this effectively forms a BU with four branches [32] but at double the cost. The use of 4-branch BUs is not common, in part because they are less stable than 3-branch BUs.

B. Problem formulation

We formulate our optimal path design for submarine cable systems with trunk-and-branch tree topology as a constrained SMT problem. The edges of the SMT here are minimal cost geodesics in \( \mathbb{M} \), with the cost given by Eq. (1), rather than straight lines. The problem in this paper is based on a manifold model, and the cable path between pairs of terminals is obtained by the Fast Marching method. So optimal connections between nodes, that is, geodesics, in this paper are curved.

To formulate this problem in a form useful for computation, we define \( \mathbf{H} \) as the set of mesh points in \( \mathbb{M} \), and \( \mathbf{H} \) is compact; \( N \) as the number of terminal nodes, \( \mathbf{R} \) as the set of latency constraints; \( N_b \) as the number of Steiner nodes; \( c_b \) as the cost of a BU; \( \mathbf{S} = \{s_1, s_2, \ldots, s_{N_b}\} \) as the set of positions of Steiner nodes in \( \mathbb{M} \); \( \Gamma \) as the set of geodesic curves of the Steiner tree; and \( f(\mathbf{S}) \) as the cable laying cost of the cable system which depends on the position of Steiner nodes \( \mathbf{S} \) and geodesic curves \( \Gamma \).

\[
f(\mathbf{S}) = \sum_{\gamma_i \in \Gamma} C(\gamma_i).
\] (2)

The cable system optimization problem corresponding to a constrained SMT is then

\[
\min_{\mathbf{S} \in \mathbf{H} \setminus \Gamma} \quad F(\mathbf{S}) = f(\mathbf{S}) + c_b \cdot N_b
\]

\[\text{s.t.} \quad d_k(x_i, x_j) \leq b_k(x_i, x_j), \quad k = 1, \ldots, |R|,
\]

where \( x_i \) and \( x_j \) are any pair of terminal nodes with latency requirement; the value of the latency requirement is written as \( b(x_i, x_j) \). \( d_k(x_i, x_j) \) is the length of the cable path between nodes \( x_i \) and \( x_j \),

\[
d_k(x_i, x_j) = \sum_{\gamma_i \in \mathbf{E}_k} l(\gamma_i),
\]

where \( \mathbf{E}_k \) is the set of geodesic curves between nodes \( x_i \) and \( x_j \), \( \mathbf{E}_k = \{\gamma(x_i, s_1), \gamma(s_1, s_2), \ldots, \gamma(s_b, x_j)\} \), \( s_1 \) is the intermediate Steiner node in the path between nodes \( x_i \) and \( x_j \). In general, there may be latency constraints on several pairs of nodes.

A simple example is shown in Fig. 1 which involves four nodes \( x_1, x_2, x_3, x_4 \) and two Steiner nodes \( s_1, s_2 \). There is a constraint on the length of the path between \( x_1 \) and \( x_2 \), i.e., \( d(x_1, x_2) \leq b(x_1, x_2) \). This constrained optimization problem in this case can be formulated as follows.

**Fig. 1:** An example of the constrained SMT problem.

\[
\min_{s_1, s_2 \in \mathbf{H} \setminus \Gamma} \quad F(s_1, s_2) = f(s_1, s_2) + 2c_b
\]

\[\text{s.t.} \quad l(\gamma(x_1, s_1)) + l(\gamma(s_1, x_2)) \leq b(x_1, x_2).
\] (4)

IV. SOLUTIONS

Here we describe a method called LAFM to solve the constrained SMT problem on the manifold discussed in Section III. We first apply the Lagrangian penalty-function approach described in Everett [63] to convert the constrained problem to an unconstrained problem. Next, we introduce
the LAFM-I algorithm and prove that it can find the optimal solution for single constraint problems. Then we describe an enhancement, LAFM-II, for multiple constraint problems.

A. LAFM for a given topology

To convert the problem in Eq. (3) to an unconstrained problem, we construct the simultaneous equations by the typical Lagrange multiplier model:

\[ L(S, \lambda) = f(S) + \sum_{k=1}^{\left| R \right|} \lambda_k (g_k(S) - b_k(x_i, x_j)) + c_\lambda \cdot N_\lambda, \]

(5)

where \( \lambda = [\lambda_1, \lambda_2, \ldots, \lambda_{\left| R \right|}]^T \) is the vector of Lagrange multipliers and \( \lambda_k \geq 0 \). \( f(S) \) represents the cable laying cost of the system. \( g_k(S) \) re-denotes \( d_k \), which is determined by the position of Steiner nodes on the path between constrained nodes \( x_i \) and \( x_j \). For a cable system with a given topology and terminals, \( f(S) \) and \( g(S) \) are functions of the Steiner nodes \( S \) and the cable paths.

Before trying to find the optimal solution of the constrained problem (3), we need to eliminate the case where the optimal solution does not exist. That is, if one of the constraints, \( k_i \), is such that \( l(\gamma_k(x_i, x_j)) > b_k(x_i, x_j) \), i.e., the allowed maximum length of the path connecting \( x_i \), as well as \( x_j \) (via Steiner nodes) is less than the length of the geodesic curve between them (whether or not it is via Steiner nodes), then there is no optimal solution for the problem (3). We make the assumption that this is not the case, and therefore, that an optimal solution of the problem (3) exists.

Firstly, we define \( m^*_I(\lambda) \) as

\[ m^*_I(\lambda) = \min_{S \in H_I} L(S, \lambda). \]

(6)

Then the dual problem to (5) can be formulated as

\[ \max \min_{\lambda \in \Lambda} L(S, \lambda) = \max_{\lambda} m^*_I(\lambda). \]

(7)

By weak duality, for any given \( \lambda \geq 0 \), \( m^*_I(\lambda) \) provides a lower bound on the optimal value of the problem (3) if \( S \) is feasible with respect to the constraints in problem (3).

In [63], it is concluded that the optimal solution of the primal problem (3) is equal to the solution to the dual problem (7) if the latter satisfies the complementary slackness condition, namely \( \lambda_k g_k(S^*) = \lambda_k b_k(x_i, x_j), k = 1, 2, \ldots, \left| R \right| \), where \( S^* \) solves the dual problem for \( \lambda = \lambda^* \).

We first consider the SMT problem in the case of one latency constraint. For this, we propose the LAFM-I algorithm in Section IV-B. Under mild assumptions, we prove that LAFM-I is able to find the optimal solution of the primal problem (3) in Section IV-C. Turning to the SMT problem with more than one latency constraint, we propose the LAFM-II algorithm in Section IV-D. We show that LAFM-II can, under some circumstances, find an optimal solution to the primal problem (3). Where we cannot guarantee optimality, LAFM-II provides a reasonably good suboptimal solution.

B. LAFM method for one latency constraint problem

We first describe the LAFM-I algorithm; that is, with one latency constraint. As shown in Algorithm 1, LAFM-I can be regarded as an extension of the dynamic programming algorithm for a DAG-Least-Cost-System in [7, 32] used to find the SMT without any latency constraints on a triangulation irregular 2D manifold in a 3D space for a given topology. LAFM-I obtains \( S^* \) by finding the minimum value of \( L(S, \lambda) \) in Eq. (5) for a fixed \( \lambda \) at every iteration. On the premise of \( l(\gamma_k(x_i, x_j)) \leq b \), LAFM-I starts with \( \lambda \) equal to a large number, leading to a feasible solution, then updates \( \lambda \) along the lines of the primal-dual subgradient method [64]. More specifically, at iteration \( I \), \( \lambda^I \) is updated by

\[ \lambda^I+1 = \lambda^I + \frac{1}{I} \cdot \frac{g(S^I) - b}{g(S^I) - b}, \]

if \( g(S^I) - b \neq 0 \),

(8)

where \( b \) is the allowed maximum latency, \( g(s^I) \) is smaller than \( b \) when LAFM-I starts with \( \lambda \) large enough to lead to a feasible solution. The value of \( \lambda \) decreases from one iteration to the next, and, as mentioned before, \( g(s) \) increases with decreasing \( \lambda \), whereas \( f(s) \) decreases. When \( g(s) \) takes the maximum value that satisfies the constraints, \( f(s) \) takes the minimum value, and this is the optimal solution that we seek.

C. Optimality of LAFM-I

To prove that LAFM-I finds the optimal solution of the primal problem (3), firstly we write:

\[ M^{\ast}_\lambda = \{ S : L(S, \lambda) = m^*_I(\lambda) \} \]

\[ m^*_I(\lambda) = \min \{ f(S) : L(S, \lambda) = m^*_I(\lambda) \} \]

\[ m^{\ast}_I(\lambda) = \max \{ g(S) : L(S, \lambda) = m^*_I(\lambda) \}. \]

We make the following assumptions. The first is a reasonable one and will generally be true in practice.

Assumption 1. Both \( f(S) \) and \( g(S) \) are continuous in \( S \).

The second is more problematic, but is, at this time, required for us to prove the result. We believe that some version of the result continues to be true in more generality but we have been unable to prove it. We hope to return to this in a later paper.

Assumption 2. For each value of \( \lambda \), \( |M^{\ast}_\lambda| = 1 \).

The following theorem proves the optimality of LAFM-I.

Theorem 1. For a given topology, under Assumptions 1 and 2, LAFM-I can provide an optimal solution for cable system design taking account of cable laying cost, BU cost, as well as a latency constraint for one pair of terminal nodes.

The proof of Theorem 1 relies on the following Lemma 1.

Lemma 1. For a fixed given topology, and one latency requirement, \( m^*_I(\lambda) \) is non-decreasing in \( \lambda \) and \( m^{\ast}_I(\lambda) \) is continuous and non-increasing in \( \lambda \).

See Appendix I for the proof of Lemma 1.
Algorithm 1 The LAFM-I Algorithm.

Input:
Coordinates of N terminal nodes, the constrained pair of nodes (x_i, x_j), the constraint requirement b, cable laying cost c, BU construction cost c_o, and the topology of Steiner tree.

Output:
Coordinates of Steiner nodes S* = {s_i, i = 1, 2, ..., m - 2}, and the path of the Steiner tree, Γ.

1: Run FMM for each terminal node x_i, i = 1, 2, ..., n, to obtain the corresponding level set (distance map) d_i.
2: Obtain the coordinates of Steiner nodes S by executing the DAG-Least-Cost-System algorithm [7, 32] with input of distance map d_i and the given topology.
3: if (l(γ(x_i,x_j)) - b) < 0 then
4: return t;
5: else
6: Let λ^0 be a large number which leads to a feasible solution;
7: Update the distance map d_i according to the value of the λ associated with the constraint:
   d_{x_i} = (1 + λ) · d_{x_i}, d_{x_j} = (1 + λ) · d_{x_j},
   where x_i and x_j are the constrained pair of nodes.
8: Obtain the coordinates of Steiner nodes S by executing the DAG-Least-Cost-System algorithm [7, 32] with the updated d_i and the given topology.
9: while g(S) - b ≠ 0 do
10:   λ^t+1 = λ^t + 1 · g(S^t) - b;
11:   Update the distance map d_i with new λ;
12:   Obtain the coordinates of Steiner nodes S by executing the DAG-Least-Cost-System algorithm [7, 32] with the updated d_i and the given topology.
13: I = I + 1;
14: end while
15: end if
16: Obtain the optimal result with Steiner nodes S*;
17: Calculate the minimum cost T for the given topology,
   T = f(S*) + N_b · c_o;
18: return S, Γ and T.

Proof. Proof of Theorem 1. The monotonicity of m_y^*(λ) and m_y(λ) is illustrated in Fig. 2. For a given latency requirement b and l(x_i,x_j) ≤ b, we start with a large λ in Algorithm 1, and adjust λ according to Eq. (8). By the monotonicity and continuity of m_y^*(λ), we can find such a λ^* for which g(S*) = b. Then, the corresponding set of Steiner nodes S*, for λ = λ^*, is the optimal solution of the primal problem (3) since it satisfies the complementary slackness condition.

Remarks:
• Because of the limited accuracy of terrain data sampling, we cannot guarantee to find a path with a length equal to the value of the latency constraint. In practice, we may stop the iteration in Algorithm 1 once the length of the constrained path is very close to the latency requirement.
• The optimality proof of LAFM-I is based on Assumptions 1 and 2. Assumption 1 is not particularly demanding. However, Assumption 2, which is a key condition for the continuity of m_y^*(λ), is not straightforward and is indeed not always true. Fig. 3 shows an example where |M| = 2. Without Assumption 2, m_y^*(λ) would be semi-continuous [65]. Under this circumstance, the problem of determining in which scenarios LAFM-I can find S* such that g(S*) = b is much deeper and we leave it to our future work. In practice, at least we can keep the least upper bound on f(S) in the iteration and output it as a suboptimal solution of the primal problem (3).

D. LAFM method for multiple latency constraints problem

Problems with more than one latency requirement are much more complicated. Here we propose another algorithm, called LAFM-II, described in Algorithm 2, for multiple constraint problems. LAFM-II is an extension of LAFM-I to the multiple λs case. To begin with, for the given topology, we calculate and determine the relationship between l_k(x_i,x_j) and b_k. If one of l_k(x_i,x_j) > b_k, we can say that there is no feasible solution for this topology. On the premise of l_k(x_i,x_j) ≤ b_k, k = 1, 2, ..., |R| for a given topology, LAFM-II starts with λs equal to values large enough to lead to a feasible solution, S', and its corresponding f_{LB}(S'). Then we update all λs using Eq. (9).

λ_k^{t+1} = λ_k^t + \frac{1}{I} · \frac{g_k(S^t) - b_k}{|g_k(S^t) - b_k|}, if g_k(S^t) ≠ b_k. (9)
After every iteration, \( S' \) and \( f_{LB}(S') \) will be updated if the current \( f(S) \) is less and the constraints are satisfied. LAFM-II stops when \( \lambda_k g_k(S) = \lambda_k b_k, \ k = 1, 2, \ldots, |R| \), or the maximum number of iterations \( I_{\text{max}} \) is reached.

From the above description of LAFM-II, we conclude that, if LAFM-II outputs a solution within a predefined maximum number of iterations, then this solution is the optimal solution of the primal problem (3) since the complementary slackness condition is satisfied. However, in general, LAFM-II cannot guarantee to find the optimal solution of the primal problem (3) because of its complicated nature. If the 2D manifold is simply the 2D plane in Problem (3), the conclusion may be strengthened since both \( f(s) \) and \( g_k(s) \) are convex. Our simulations show that, in many cases, LAFM-II indeed obtains the optimal solution of problem (3) as the examples shown in Section V-B. In addition, we can keep the best bound for \( f(s) \) in the iteration of LAFM-II and use it as an estimate of the optimal \( f(s) \) if LAFM-II does not find the optimal one. The gap, denoted as \( G_{LU} \), between the upper and lower bounds of the optimal solution depends on the Lagrange multipliers and the difference from the real length of constraint path to the length requirement as shown in Eq. (10). We leave the theoretical investigation of Problem (3) in the case of multiple constraints for future work.

\[
G_{LU} = \sum_{k=1}^{|R|} \lambda_k (g_k(S) - b_k(x_i, x_j)). \tag{10}
\]

E. LAFM method for unknown topology

The above discussion is for a set of terminal nodes with a given topology. The LAFM-I and LAFM-II algorithms start with a known topology. Based on this topology, we know the number of the BUs and how these BUs connect with each other and terminals. We also know the traffic requirements through each BU, and these enable accurate estimation of the cost.

There is nothing in the algorithm that prohibits use of different costs for different BUs. For an unknown topology problem, we need to consider all possible topologies for terminal nodes, including all full Steiner topologies, including degenerate topologies. This is because, when considering the cost of BU, the full Steiner topologies with minimal total cable laying cost is not exactly the minimal cable system construction cost. The increase in the total cable length may result in a decrease in the number of BUs and hence possibly the total cost. Based on engineering experience, we try to avoid cross-laying of cables, by this way, the number of possible topologies will be decreased. For multiple latency requirements problems, the possible topologies for these terminals are easier to eliminate, so simplifying the selection process of the topological structure and reducing the amount of calculation.

We list the process as follows.

- List all possible full Steiner topologies in the 2D manifold by Smith’s BkE method [35]. We note that all degenerate topologies can be derived from the full Steiner topologies [32].
- For every full Steiner topology, run the Dijkstra's algorithm to find the internal nodes along the path between constraint pairs of nodes.

Algorithm 2 The LAFM-II Algorithm.

**Input:**

The coordinates of \( N \) terminal nodes, the set of constraint requirements \( R = \{r_k, k=1,2,\ldots,|R|\} \) which illustrate some pairs of nodes with latency requirements \( b_k \), cable laying cost \( c \), BU construction cost \( c_o \), and the topology of Steiner tree.

**Output:**

Coordinates of Steiner nodes \( S = \{s_i, i=1,2,\ldots,n-2\} \) and the path of the Steiner tree, \( \Gamma \).

1. Run FMM for each terminal node \( x_i, i=1,2,\ldots,n-2 \), and the path of the Steiner tree, \( \Gamma \).
2. Obtain the coordinates of Steiner nodes \( S \) by executing the DAG-Least-Cost-System algorithm with the distance map \( d_i \) and the given topology.
3. For each constraint requirement \( r_k \) do
   a. if \( (l|\gamma_k(x_i, x_j)| - b_k) < 0 \) then
      return \( \emptyset \);
   end if
4. Let \( x_i \) be one terminal of constraint \( r_k \) then
   a. \( \bar{\lambda} = \lambda + \lambda_k \),
   end if
5. \( d_{x_i} = (1 + \bar{\lambda}) \cdot d_{x_i} \),
6. end for
7. Obtain the coordinates of Steiner nodes \( S \) and corresponding \( f_{LB}(S') \) by executing the DAG-Least-Cost-System algorithm with the distance map \( d_i \) and the given topology.
8. while \( \lambda_k \cdot g_k(S') = \lambda_k \cdot b_k & I \leq I_{\text{max}} \) do
   a. \( \lambda^{I+1} = \lambda^I + \frac{1}{\gamma_k(S') - b_k}, k = 1,\ldots,|R|; \)
   b. if \( f(S') < f_{LB}(S') \) then
      \( f_{LB}(S') = f(S'); \)
   end if
9. Update the distance map \( d_i \) according to the value of the \( \bar{\lambda} \)s associated with the constraints;
10. for \( i = 1,2,\ldots,n \), do
   a. \( \bar{\lambda} = 0 \),
   end for
11. for \( k = 1,2,\ldots,|R| \) do
   a. if \( x_i \) is one terminal of constraint \( r_k \) then
      \( \bar{\lambda} = \lambda + \lambda_k \),
   end if
12. \( d_{x_i} = (1 + \bar{\lambda}) \cdot d_{x_i} \),
13. end for
14. Obtain the coordinates of Steiner nodes \( S \) and the corresponding \( f_{LB}(S') \) by executing the DAG-Least-Cost-System algorithm with the updated \( d_i \) and the given topology.
15. while \( \lambda_k \cdot g_k(S) = \lambda_k \cdot b_k & I \leq I_{\text{max}} \) do
   a. \( \lambda^{I+1} = \lambda^I + \frac{1}{\gamma_k(S') - b_k}, k = 1,\ldots,|R|; \)
   b. if \( f(S') < f_{LB}(S') \) then
      \( f_{LB}(S') = f(S'); \)
   end if
16. Obtain the coordinates of Steiner nodes \( S \) and the corresponding \( f_{LB}(S') \) by executing the DAG-Least-Cost-System algorithm with the updated \( d_i \) and the given topology.
17. end while
18. Obtain the optimal or suboptimal result with Steiner nodes \( S' \);
19. Calculate the minimum cost \( T \) for this topology, \( T = f(S') + N_k \cdot c_o \);
20. return \( S', \Gamma \) and \( T \).
Note: when applying the Dijkstra’s algorithm, the weight of every path between two nodes is regarded as the same.

- Taking account of the number of internal nodes along the path, list all possible degenerate topologies. The detailed enumeration principles are as follows.

1) If there is one internal node along the constrained path as shown in Fig. 4(a), the latency requirement is between nodes $x_1$ and $x_5$, we can get the optimal solution by applying LAFM for this topology. When BU cost is regarded as part of the total cable construction cost, as $\lambda$ changes, the number of BUs will reduce when the position of a Steiner node coincides with one of the constrained pair of nodes. So, there are two cases as shown in Figs. 4(b)–4(c). For these two cases, the constrained pair of nodes $x_1$ and $x_5$ are already connected, so we just need to run the dynamic programming for the remaining four nodes without considering the constraint requirement.

2) If there are two internal nodes along the constrained path as shown in Fig. 5(a) then, as $\lambda$ changes, the number of BUs may reduce. There are five cases, as shown in Figs. 5(b)–5(f).

3) If there are three internal nodes along the constrained path as shown in Fig. 6(a), a similar analysis to the two internal nodes case reveals that, for three internal nodes topologies, there are eight cases as shown in Figs. 6(b)–6(i).

We conclude that, if there are $n$ internal nodes on the path between constrained pair of nodes, there will be $3^n$ different topologies, including the original full Steiner case, because of the various combinations of Steiner nodes and terminal nodes. We observe that the topologies in (d), (e), and (f) in Fig. 5 do not need to be considered because they will inevitably lead to solutions that are no better than the topologies in Fig. 4.

Note that, when the constrained pair of nodes $x_1$ and $x_5$ are directly connected, we just need to apply the FMM method to optimize the path between $x_1$ and $x_5$, and run the dynamic programming algorithm for DAG-Least-Cost-System for the remaining four nodes without considering the constraint requirement.

For each topology, we combine the LAFM method, the dynamic programming algorithm for DAG-Least-Cost-System, and the FMM method (for point-to-point distance calculation) to calculate the total cost of the cable system and choose the smallest one.

Fig. 4: The SMT with one internal node along the constrained path and its degenerated cases.

Fig. 5: The SMT with two internal nodes along the constrained path and its degenerated cases.

Fig. 6: The SMT with three internal nodes along the constrained path and its degenerated cases.

F. Computational complexity analysis of LAFM

As the SMT problem is NP-hard, we just derive the computational requirement of one iteration for a given topology. We assume $|H|$ grid nodes on the area $M$ and $N$ terminals to be connected. The complexity analysis is then the same as in [7, 32]: for a given topology with $N_h$ Steiner nodes, the computational complexity of every iteration is $O\left(|H|^2 \log |H| + N_h - 1\right)$ [7]. The running time of the LAFM algorithm depends on the setting of the initial value of $\lambda$ and the allowable tolerance range resolution. The cost for each pair of grid nodes on the manifold M is computed and stored in a database. Therefore, in the worst case, the algorithm requires at least $O\left(|H|^2\right)$ of memory.
V. NUMERICAL RESULTS

In this section, we apply our method to three realistic examples using bathymetric data with 30 arc-second increments in longitude and latitude from the Global Multi-Resolution Topography synthesis [66]. The object region $D$ for the three examples spans from the northwest corner ($45.000^\circ$N, $0.000^\circ$E) to the southeast corner ($36.000^\circ$N, $11.000^\circ$E). There are five terminal nodes in this area as shown in Fig. 7. Marseille ($43.297^\circ$N, $5.359^\circ$E), Sardegna ($40.557^\circ$N, $8.312^\circ$E), Algiers ($36.761^\circ$N, $3.074^\circ$E), Barcelona ($41.368^\circ$N, $2.190^\circ$E) and Annaba ($36.928^\circ$N, $7.760^\circ$E), which are denoted as $A$, $B$, $C$, $D$, $E$, respectively. We consider a BU cost to be $1 million.

As mentioned in the Introduction, ICPC [6] recommends that the length of the cable should be as short as possible in shallow water, except in the area close to the cable landing station. The cable should preferably be laid deep in the sea, thereby, reducing the risks posed by human activities. As the depth of cable laying areas increases, the threat of cable failure decreases rapidly [67]. The design of Eq. (11) and the selection of cable laying areas increases, the threat of cable failure decreases rapidly [67]. The design of Eq. (11) and the selection of parameters are based on a cable engineer’s suggestion.

Note that this equation is not absolute and can be adjusted according to different actual situations. This has no impact on the application of our algorithm.

We consider the depth, denoted by $z$, considered in our objective cost function in units of USD $\$. 

$$c(x) = \begin{cases} 37500, & \text{if } z \geq 0 \text{ km}, \\ 25000 - 25000 \times |z|, & \text{if } 0 \text{ km} < z \geq -0.2 \text{ km}, \\ 25000, & \text{otherwise} \end{cases}$$

(11)

where $x$ is a mesh node and $c(x)$ is the cost of cable per km at this mesh node. We ran the LAFM-I and LAFM-II algorithms for all possible solutions, according to whether the problem involved a single latency constraint or multiple latency constraints, based on a resolution of about 7.48 km, to decide the location of the Steiner nodes and the cable path. For comparison, we applied SA to design submarine cable paths under a single latency constraint as well as multiple latency constraints based on the same resolution. See Appendix II for the details of SA. For a single constraint problem, as LAFM-I is proved to be optimal, we can use it as a benchmark to assess the performance of SA. Then, for multiple constraint problems, we compare the results of the two approaches and choose the better result. We have demonstrated two cases of two and four latency constraints where LAFM-II can provide optimal solutions, and have shown that LAFM-II outperforms SA in these two cases. The experiments based on the LAFM algorithms in this section are run on a platform with a 2.00GHz Intel Core i5 CPU while the experiments based on SA are run on a platform with a 2.60GHz Intel Core i7-9750H CPU.

A. Example with one latency requirement

We aim to find a cable system at minimum cost connecting the five terminal nodes in Fig. 7, where $\text{cost}$ is defined as the total cost of the cable system including BU construction cost and cable laying cost, and there is a latency requirement between nodes $A$ and $E$ ($\lambda_{A,E} \leq 500$ km).
Fig. 9: (a) the optimization process and (b) the optimal SMT result of LAFM-I algorithm for Topology I.

Fig. 10: The degenerated cases of Topology I with one Steiner node along the constrained path.

degenerate topologies in which nodes A and E are connected directly:

- A cable system without constraint consists of nodes A, B, C, D; and a cable path connects A and E directly, as shown in Fig. 11(a);
- A cable system without constraint consists of nodes E, B, C, D, and a cable path connects A and E directly, as shown in Fig. 11(b);

Fig. 11: The results from degenerated cases of Topology I without Steiner node along the constrained path.

For Topology II shown in Fig. 8(b), we compare the corresponding six cases shown in Figs. 12(a)-12(f). Fig. 12(a) provides the optimal result for the case where there are three BUs along the path between nodes A and E; Figs. 12(b) and 12(c) provide the optimal results for the two cases where there are two BUs along the path between nodes A and E. As one of the Steiner nodes is driven to coincide with node A (in the first case) or with node E (in the second case), this case is reduced to the case shown in Figs. 12(d)-12(f), thus providing an optimal result with only one BU along the path between nodes A and E. For the case where there is no BU along the path between nodes A and E, the results correspond to the topologies in Fig. 11(a)-11(b).

Fig. 12: The results (a) from Topology II and (b)-(f) its degenerated cases.

For Topology III shown in Fig. 8(c), the degenerate topologies with one Steiner node along the constraint path are shown in Figs. 13(a)-13(b).

Fig. 13: The results (a) from Topology III and (b) its degenerated cases.

The final best results for Topology IV and Topology V shown in Figs. 8(d) and 8(e) are presented in Figs. 14(a) and 14(b). The resulting cases forming the degenerate cases of these two topologies can be found in the degenerate cases for Topologies I-III discussed above.

Finally, the results generated from the five possible full Steiner topologies are included in Table I. By comparing all these results, we obtain the optimal outcome for this example as shown in Fig. 14(a), with a total cost of $9.61 million and two BUs located at (41.5887°N, 4.8801°E) and (40.0425°N, 7.2556°E).

B. Multiple latency requirements

In this subsection, we apply LAFM-II to solve the optimization problem for a cable system with multiple constraints.
TABLE I: The SMT results of LAFM-I for different topologies.

<table>
<thead>
<tr>
<th>Topology</th>
<th>BU number along the path</th>
<th>$l_{AE}$ (km)</th>
<th>BU number in the system</th>
<th>total length (km)</th>
<th>total cost (million $)</th>
<th>Figure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology I</td>
<td>2</td>
<td>482.62</td>
<td>3</td>
<td>1781.09</td>
<td>9.72</td>
<td>Fig. 9(b)</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>487.08</td>
<td>2</td>
<td>1769.95</td>
<td>10.13</td>
<td>Fig. 10(a)</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>485.20</td>
<td>2</td>
<td>2006.86</td>
<td>11.08</td>
<td>Fig. 10(b)</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>490.22</td>
<td>2</td>
<td>1876.54</td>
<td>9.96</td>
<td>Fig. 11(a)</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>490.22</td>
<td>2</td>
<td>1900.71</td>
<td>10.54</td>
<td>Fig. 11(b)</td>
</tr>
<tr>
<td>Topology II</td>
<td>3</td>
<td>488.87</td>
<td>3</td>
<td>1900.84</td>
<td>11.32</td>
<td>Fig. 12(a)</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>494.76</td>
<td>2</td>
<td>2450.20</td>
<td>11.17</td>
<td>Fig. 12(b)</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>485.20</td>
<td>2</td>
<td>2006.86</td>
<td>10.68</td>
<td>Fig. 12(c)</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>498.43</td>
<td>1</td>
<td>2565.55</td>
<td>10.54</td>
<td>Fig. 12(d)</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>488.21</td>
<td>1</td>
<td>2228.17</td>
<td>10.33</td>
<td>Fig. 12(e)</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>473.98</td>
<td>1</td>
<td>2179.89</td>
<td>10.04</td>
<td>Fig. 12(f)</td>
</tr>
<tr>
<td>Topology III</td>
<td>1</td>
<td>485.20</td>
<td>1</td>
<td>1802.11</td>
<td>10.26</td>
<td>Fig. 13(a)</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>490.22</td>
<td>0</td>
<td>1906.79</td>
<td>9.75</td>
<td>Fig. 13(b)</td>
</tr>
<tr>
<td>Topology IV</td>
<td>1</td>
<td>446.69</td>
<td>2</td>
<td>1670.37</td>
<td>9.6</td>
<td>Fig. 8(d)</td>
</tr>
<tr>
<td>Topology V</td>
<td>1</td>
<td>482.89</td>
<td>1</td>
<td>1950.38</td>
<td>9.67</td>
<td>Fig. 8(e)</td>
</tr>
</tbody>
</table>

Fig. 14: Results from Topology IV and Topology V.

Firstly, we aim to find a Steiner tree with minimum cost to connect the five terminal nodes, satisfying two latency requirements between two pairs of nodes, (A, E) and (B, C), with the constraints $l_{A,E} \leq 450$ km and $l_{B,C} \leq 510$ km, respectively.

For multiple latency requirements problems, the possible topologies for these terminals are easier to eliminate, so simplifying the selection process of the topological structure and reducing the amount of calculation. For the five full Steiner topologies shown in Fig. 8, the optimal result is generated from the topologies in Fig. 8(d) or Fig. 8(e).

The process of obtaining the optimal solution based on the topology in Fig. 8(d) is shown in Fig. 15(b). LAFM-II begins at a feasible but not optimal solution with $\lambda_1 = 10$ and $\lambda_2 = 10$. Then, $\lambda_1$ and $\lambda_2$ are updated in every iteration according to Eq. (8). LAFM-II stops when $\lambda_1 = 0.735$ and $\lambda_2 = 0$. The length of the cable path between nodes A and E is 446.699 km which can be regarded the same as the required length considering the resolution of 7.48 km. The length of the cable path between nodes B and C is 464.433 km which satisfies the requirement with $\lambda_2 = 0$. In accord with the discussion in Section IV-D, we regard this solution as an approximate optimal solution ($G_{I,u} = 0$), resulting in a total cost of $9.61$ million and a total length of 1670.373 km. It takes about 4.5 hours to execute LAFM-II for this topology. The process of getting the optimal solution based on the topology in Fig. 8(e) is shown in Fig. 15(a). LAFM-II stops when $I > I_{max}$ ($I_{max} = 100$) resulting in a total cost of $10.83$ million and a total length of 1873.13 km. It takes about 6.5 hours to execute LAFM-II for this topology.

After comparison, the final result for this two constraint problem is the one based on the topology in Fig. 8(d) as shown in Fig. 15(c) with two BUs located at (41.5887°N, 4.8801°E) and (40.0425°N, 7.2556°E). Table II shows the details of the approximate optimal solution.

We then aim to find a Steiner tree with minimum cost to connect these five terminal nodes, satisfying latency requirements between four pairs of nodes, (A, E), (A, B), (B, C) and (B, D) with $l_{A,E} \leq 450$ km, $l_{A,B} \leq 500$ km, $l_{B,C} \leq 500$ km and $l_{B,D} \leq 700$ km, respectively. After excluding topologies that clearly cannot satisfy the constraints, we find that the topology shown in Fig. 8(d) provides a solution that satisfies the constraints and achieves the shortest length among solutions considered by our algorithms. The performance of LAFM-II depends on the initial value. We initialise the value of $\lambda$ with $\lambda_1 = 10$, $\lambda_2 = 10$, $\lambda_3 = 10$, $\lambda_4 = 10$ and then update them according to Eq. (9). LAFM-II stops when $\lambda_1 = 5.686$, $\lambda_2 = 0$, $\lambda_3 = 4.102$, $\lambda_4 = 1.068$, resulting in an approximate optimal solution ($G_{I,u} = 0$) with...
### C. Comparison with SA

Here, we apply the SA algorithm to design submarine cable paths under various of latency requirements which were achieved by LAFM previously. SA is a well-known probabilistic technique for approximating the global optimum of a given function. Specifically, it provides an approximation to the global optimum in a large search space. For problems where finding an approximate global optimum is more important than finding a precise local optimum in a fixed amount of time, SA can be preferable to exact algorithms such as gradient descent or branch and bound. SA has been widely used to provide high-quality solutions in an extensive solution space and NP-hard combinatorial problems [68]. In the context of cable path planning, SA has been used in Wang et al. [5] to obtain a high-quality cable path that is significantly better than a real-world path, designed manually by a cable expert. Since the problem of finding the optimal positions of several BUs in a huge discrete mesh network with billion of nodes is an NP-hard problem, to shorten the solution time, we use SA to find the optimal solution and compare the results obtained by LAFM to show the performance of the LAFM method. Table IV provides the parameter settings of SA. Table V shows the solutions obtained by SA under one, two, and four constraints of the given topologies shown in Fig. 8(d).

For SA, the total cost and length for the problem with one latency constraint are $10.32 million and 1689.48 km, respectively. Recall that the total cost and length we obtained by LAFM-I were $9.61 million and 1670.37 km, respectively. As we know that LAFM-I is optimal, SA cannot perform better than LAFM-I. For the problem with two latency constraints, the total cost and the total length obtained by SA are $9.68 million and 1744.68 km, respectively, while LAFM-II found the optimal solution. For the case of four latency constraints, the total cost and length obtained by SA are $10.70 million and 1798.07 km, which are again worse than the optimal results obtained by LAFM-II.

The running times for SA are in the range of 20 – 26 hours for each example, as shown in Table V. The running times of LAFM-I and LAFM-II depend on the initial value and update strategy of λs. For the examples in this section, the average running times for the LAFM method were in the range of 4.5 – 6.5 hours, as discussed previously in this section which are significantly shorter than the running times of SA.

Through the comparison of LAFM and SA both in terms of the quality of the results and the running times above, we have demonstrated the superiority of our LAFM method for the submarine cable network optimization with latency constraint requirements.

### VI. CONCLUSION

We have provided a method, called LAFM, for optimizing a trunk-and-branch tree topology cable system with latency constraints. This has been done for a network on the surface of the earth modeled by a triangulated manifold, taking into account the total cost of the cable system, which includes both actual cable and BU costs and a measure of location-dependent risk. Specifically, we focus on minimizing the cost of the cable system while ensuring that cable length between some pairs of nodes satisfies latency requirements. Our method is based on FMM to find the optimal cable path between any pair of locations (mesh point) in the area and then determine the location of the Steiner nodes. The Steiner nodes in our problem can vary in number while incurring a penalty (cost), and their degrees are constrained to three. We have proved that our method (LAFM-I) can find the optimal solution for cable systems with one latency constraint and a feasible solution for cable systems with multiple latency constraints. In addition to this feasible solution that serves as an upper bound to the optimal cost solution, we have also provided a lower bound for the optimal cost solution. We have shown that LAFM-II can provide a solution with provable bounds for problems with multiple latency constraints. We have also demonstrated finding optimal solutions (zero gap between the bounds) for examples with two and four latency constraints. In addition, using numerical results based on realistic scenarios, we have demonstrated the superiority of our LAFM method over SA. Furthermore, in cases where LAFM-II is not optimal, combining LAFM-II and SA, by using the result of the LAFM-II to seed the SA algorithm, can provide further improvements where the result of LAFM-II is not optimal.

### APPENDIX I

#### PROOF OF LEMMA 1

**Proof.** First consider the case $\lambda_0 = 0$, $\lambda_1 \geq 0$, and let the location of the Steiner nodes based on the given topology be

<table>
<thead>
<tr>
<th>Topology</th>
<th>Requirements</th>
<th>Constrained length</th>
<th>$\lambda$</th>
<th>BU number</th>
<th>total length</th>
<th>total cost (million $)</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology IV</td>
<td>$l_{AE} \leq 450$ km, $l_{BC} \leq 510$ km</td>
<td>$l_{AE} = 446.69$ km, $l_{BC} = 464.43$ km</td>
<td>$\lambda_1 = 0.735$, $\lambda_2 = 0$</td>
<td>2</td>
<td>1670.37 km</td>
<td>9.61</td>
<td>Fig. 15(c)</td>
</tr>
</tbody>
</table>
TABLE III: The final SMT result of LAFM-II with four constraints.

<table>
<thead>
<tr>
<th>Topology</th>
<th>Requirements</th>
<th>Constrained length</th>
<th>$\lambda_s$</th>
<th>BU number</th>
<th>total length</th>
<th>total cost (million $)</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology IV</td>
<td>$l_{AE} \leq 450$ km</td>
<td>$l_{AE} = 455.26$ km</td>
<td>$\lambda_1 = 5.686$</td>
<td>3</td>
<td>1682.75 km</td>
<td>10.18</td>
<td>Fig. 16</td>
</tr>
<tr>
<td></td>
<td>$l_{AB} \leq 500$ km</td>
<td>$l_{AB} = 416.92$ km</td>
<td>$\lambda_2 = 0$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$l_{BC} \leq 500$ km</td>
<td>$l_{BC} = 500.46$ km</td>
<td>$\lambda_3 = 4.102$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$l_{BD} \leq 700$ km</td>
<td>$l_{BD} = 698.78$ km</td>
<td>$\lambda_4 = 1.068$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TABLE IV: Parameter setting of SA.

<table>
<thead>
<tr>
<th>Latency requirement(s)</th>
<th>Initial temperature $T_0$</th>
<th>Annealing temperature $T_k$</th>
<th>Terminal temperature $T_f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$l_{AE} \leq 500$ km</td>
<td>500</td>
<td>$T(k) = \frac{T_0}{1 + \log(k)}$</td>
<td>0</td>
</tr>
</tbody>
</table>

TABLE V: Results of SA under different latency constraints.

<table>
<thead>
<tr>
<th>Latency requirement(s)</th>
<th>Total length (km)</th>
<th>Total cost (million $)</th>
<th>Constrained length (km)</th>
<th>Running time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$l_{AE} \leq 500$ km</td>
<td>1689.48</td>
<td>10.32</td>
<td>$l_{AE} = 478.86$</td>
<td>74772</td>
</tr>
<tr>
<td>$l_{BC} \leq 510$ km</td>
<td>1744.68</td>
<td>9.68</td>
<td>$l_{AB} = 441.19$</td>
<td>69773</td>
</tr>
</tbody>
</table>

950 $S_\lambda$, the number of the Steiner nodes is a fixed value (we ignore
951 the BUs for now). $S_\lambda$ is restricted to the mesh points on the
952 desecrated triangulated manifold, so the $S_\lambda$ is desecrated. We
953 assume a unique minimum for simplicity, but a non-unique
954 minimum can be handled with a little added burden.

\[
\begin{cases}
S_{\lambda_0} \in \text{argmin}_S f(S) \\
S_{\lambda_1} \in \text{argmin}_S f(S) + \lambda_1 g(S).
\end{cases}
\]

(Eq. 12)

957 Evidently,

\[f(S_{\lambda_0}) \leq f(S_{\lambda_1}).\]

(Eq. 13)

958 If

\[g(S_{\lambda_0}) < g(S_{\lambda_1}),\]

(Eq. 14)

959 we would have

\[f(S_{\lambda_0}) + \lambda_1 \cdot g(S_{\lambda_0}) < f(S_{\lambda_1}) + \lambda_1 \cdot g(S_{\lambda_1}).\]

(Eq. 15)

Since,

\[S_{\lambda_1} \in \text{argmin}_S \{L(S, S_{\lambda_1})\},\]

it follows that $L(S_{\lambda_1}, \lambda_1) \leq L(S_{\lambda_0}, \lambda_0)$. This contradicts

960 Eq. (15), and so

\[g(S_{\lambda_0}) \geq g(S_{\lambda_1}).\]

(Eq. 16)

963 This, combined with Eq. (13), gives the conclusions of
964 Lemma 1 in this special case. That is, $m_\lambda^*(\lambda_1) \leq m_\lambda^*(\lambda_0) =
965 m_\lambda^*(0)$, and $m_\lambda^*(0) = m_\lambda^*(\lambda_0) \leq m_\lambda^*(\lambda_1)$.

966 Now we consider the general case, $\lambda_1 > \lambda_0 > 0$. We write
967 $\mu = \lambda_1 - \lambda_0$, and

\[\tilde{f}(S) = f(S) + \lambda_0 \cdot g(S).
\]

(Eq. 17)

970 Choose

\[
\begin{cases}
S_0^\ast \in \text{argmin}_S \tilde{f}(S); \\
S_\mu^\ast \in \text{argmin}_S \tilde{f}(S) + \mu g(S),
\end{cases}
\]

(Eq. 18)

974 so that $f(S_0^\ast) = m_\lambda^*(\lambda_0)$ and $f(S_\mu^\ast) = m_\lambda^*(\lambda_1)$.

975 From the result for the special case, with $\mu$ replacing $\lambda$ and
976 $\tilde{f}$ replacing $f$,

\[\tilde{f}(S_0^\ast) \leq \tilde{f}(S_\mu^\ast),\]

and so,

\[f(S_0^\ast) + \lambda_0 \cdot g(S_0^\ast) \leq f(S_\mu^\ast) + \lambda_0 \cdot g(S_\mu^\ast),\]

(Eq. 19)

979 In other words, we have,

\[f(S_{\lambda_0}) + \lambda_0 \cdot g(S_{\lambda_0}) \leq f(S_{\lambda_1}) + \lambda_0 \cdot g(S_{\lambda_1}),\]

(Eq. 20)

983 and from (19), we know,

\[g(S_{\lambda_0}) \geq g(S_{\lambda_1}).\]

(Eq. 21)

986 It follows that

\[f(S_{\lambda_0}) \leq f(S_{\lambda_1}).\]

(Eq. 22)

989 These yield, in the general case,

\[m_\lambda^*(\lambda_1) \leq m_\lambda^*(\lambda_0),\]

(Eq. 23)

992 $m_\lambda^*(\lambda_0) \leq m_\lambda^*(\lambda_1),\]

(Eq. 24)

995 completing the proof of monotonicity of $m_\lambda^*(\lambda)$ and $m_\lambda^*(\lambda)$
998 in Lemma 1. The continuity of $m_\lambda^*(s)$ is guaranteed by Assumption 1 and Assumption 2 according to Berge’s Maximum theorem [69, 70].

APPENDIX II

THE SA ALGORITHM

For comparison, we use simulated annealing to optimize submarine cable paths under latency requirements. Under the premiss of a given topology, we use SA to obtain the positions $B_0, B_1 := \{B_m(x_m, y_m) : m = 1, 2, \ldots, N_b\}$ of the BUs, where $x_m$ and $y_m$ are the longitude and latitude of the $m$th

BU, respectively.
SA is a technique that simulates a physical process of heating material, and then slowly lowering the temperature to reduce defects, resulting in the system falling into a minimal energy state. Kirkpatrick et al. showed that system energy can model the cost function of certain combinatorial optimization problems (e.g., the traveling salesperson problem), where seeking the lowest cost is equivalent to seeking the lowest energy [40]. Accordingly, they developed the SA algorithm based on the Metropolis algorithm [38], and used it for optimization problems. At iteration $i$, a new solution $x_{i+1}$ is generated from the current one $x_i$ according to a distribution over a neighborhood of $x_i$ that provides a tendency towards a better solution. This distribution is parameterized by the temperature parameter $T$. When $T$ is large, the distribution is more diffuse. Over the iterations, indexed by $k$, the temperature is decreased according to an annealing schedule:

$$T = T_0 > T_1 > T_2 > \cdots > T_k > \cdots$$

In typical applications, the algorithm accepts an updated solution $x_{i+1}$ if it improves the value of the objective function, but also with a certain, temperature-dependent probability, the algorithm accepts solutions that worsen the objective value. Under the right temperature schedule, accepting solutions that make the objective worse allows the algorithm to avoid falling into local minima and explore more possible solutions globally. We design an annealing schedule to lower the temperature as SA proceeds systematically. The system temperature gradually decreases as the algorithm continues, and the search range for solutions is reduced to converge to a minimum. Under certain kinds of annealing schedules and for many problems, convergence is effectively guaranteed [71, 72].

In our case, we leave the temperature constant over a number of iterations before reducing it. We then have two iteration indexes: the cooling time index $k$, and the execution index $i$. We write the temperature as $T_k$. As the SA algorithm goes on, the temperature will decrease according to $T(k) = T_0 / (1 + \log(1 + k/T_0)^{\Delta T(k)}$, with the cooling time $k$. The maximal number of iterations (indexed by $i$) at temperature $T(k)$ is denoted by $L_m^{T(k)}$.

In every iteration, one of the BU is chosen randomly and its longitude and latitude are updated. According to the new location of BUs, we calculate the current total cost $C_{current}$ and the length of the paths with latency requirements. If all the paths meet the latency requirements limitation and the current total cost $C_{current}$ is less than the best-recorded solution $B_{best}$, we update the best solution. Otherwise, if the new locations increase the total cost, these locations will be accepted with a certain probability $P = e^{-\Delta}/T(k)$ according to the Metropolis criterion [38], where $T(k)$ is the current temperature and $\Delta$ is the deviation which is equal to previous total cost minus new total cost. SA stops when the current temperature reaches $T_f$. During the execution of the SA algorithm, the slower the temperature cooling and using more iterations, the more likely it is to find the global optimal solution [39].

### Algorithm 3 The SA algorithm.

**Input:**

Given topology and $m$ initial BU locations $B_0 := \{B_m(x_m, y_m)\}$ where $x_m$ and $y_m$ are the longitude and latitude of BU $B_m$, latency constraints, initial temperature $T_0$, termination temperature $T_f$, maximal number of iterations $L_m^{T(k)}$ at temperature $T(k)$, temperature attenuation function $T(k)$ with the cooling time $k$, maximum variation for BU locations $L_{arc}$.

**Output:**

$B_{best} := \{B_m\}$ that provides paths connecting all the nodes with minimal total cost and meets all the latency requirements at the same time.

1: Connect all the nodes and BUs according to the given topology;
2: $k = 1$, $B_{current} = B_0$;
3: Calculate current total cost $C_{current}$;
4: while $T(k) > T_f$ do
5:   for $i = 1, \ldots, L_m^{T(k)}$ do
6:     for $j = 1, \ldots, m$ do
7:       $x_m = x_m + \alpha \cdot L_{arc}$;
8:       $y_m = y_m + \beta \cdot L_{arc}$;
9:   end for
10: end for
11: Connect all the nodes and BUs according to the given topology;
12: if All the paths meet the latency requirements then
13:   $B_{new} := \{B_m(x_m, y_m)\}$;
14:   Calculate new total cost $C_{new}$;
15:   if $C_{new} < C_{current}$ then
16:     $C_{current} = C_{new}$, $B_{current} = B_{new}$;
17:   else
18:     end if
19:   else
20:     $\sigma \sim U(0, 1)$;
21:     if $\sigma < e^{-(C_{new} - C_{current})/T(l)}$ then
22:       $C_{current} = C_{new}$, $B_{current} = B_{new}$;
23:     else
24:       end if
25:     $B_{new} = B_{current}$;
26:   end if
27: end if
28: end if
29: $k = k + 1$;
30: end while
31: return $B_{best}$.
REFERENCES


http://subtelforum.com/articles/products/
industry report/


