Acta Astronautica 215 (2024) 373–386

Contents lists available at ScienceDirect

Acta Astronautica

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

Research paper

Traveling Salesman Problem of optimal debris removal sequence using non-population gradient search ☆

Liqiang Hou [a,*], Arun Misra [b]

[a] School of Aeronautics and Astronautics, Shanghai Jiaotong University, 800 Dongchuan RD., Shanghai, 200240, China
[b] Department of Mechanical Engineering, Mcgill University, 817 Sherbrooke Street West, Montreal, H3A 0C3, Quebec, Canada

ARTICLE INFO

MSC:
0000
1111

Keywords:
Debris removal
Traveling Salesman Problem
Maximum likelihood
Parameter estimation

ABSTRACT

A new approach for debris removal sequence optimization with multiple spacecraft is developed and tested. The motion planning of the multiple rendezvous involves complex Traveling Salesman Problem (TSP) combinatorial decision, in which both the continuous parameters of the orbital maneuvers and integer indices of the debris are determined. Expected value and variance of trajectory conditioned on preceding visit, not the state of target itself, are propagated and analyzed. With the metric, likelihood of expected trajectory can be given, and the mixed integer problem is reformulated into continuous parameter estimation with maximum likelihood. A non-population gradient based optimizer is then used to search the target sequence. The method of TSP optimization is compared to the continuous parameter estimator of Extended Kalman Filter. Using this method, the classic static TSP-LIB benchmarks with up to 100 nodes are tested. We then extend the method to the TSP with space dynamics. Effectiveness and efficiency of the proposed method are tested by a sequential visit of near-earth debris considering J2 effects with a single and multiple spacecrafts.

1. Introduction

This paper presents a novel method for optimal debris removal sequence, by casting it as the Traveling Salesman Problem (TSP). For its industrial significance, the TSP is one of the most extensively studied computational mathematics problems. In this problem, given a set of candidate targets, e.g., major cities of a country, a sequence of visiting all the targets with the cheapest traveling cost is determined.

The mixed integer TSP can find applications in space mission design as well. The design optimization, such as multiple flybys, debris rendezvous and asteroid exploration, can be treated as the TSP with dynamics. In these problems, a set of integer-type variables represents the candidate targets. First, traveling costs between the targets such as debris, asteroid, or planets are computed using the dynamics equations. Next, one needs to determine the optimal traveling path to the targets and corresponding continuous maneuver parameters, such as time of flight and Delta-V between the targets on the path. Usually, a large number of candidates are involved in the path planning. Sometimes, the number can be up to hundreds or even higher.

Though extensively studied, there is no effective TSP solution method for the general case yet [1]. The challenge is from the intrinsical separated characteristics of the integer design variable. One must explore every possible traveling path and optimize associated continuous parameters for each pair of targets on the path. The number of combinations of paths can be huge for a brute-force strategy. Given a path planning problem of n nodes, the number of path combinations is $(n-1)!/2$. With the number of targets increasing, the number of combinatorial subspaces to be explored will soon become massive. As for the dynamic TSP of space design, the situation is more complex, considering that the traveling cost between the pairs on the path varies not only with the moving targets but also with the Time of Flight (ToF) between them.

Several algorithms are available for the TSP, e.g. Tabu search, Simulated Annealing, Genetic Algorithm, and Ant Colony Optimization [2–4]. Some are pretty efficient for the pure geometric type TSP with simple geometric distance equations, and different heuristics for path planning can be used [1]. However, few of them can be used directly to tackle the space TSP with dynamics. In most studies regarding the space TSP, the branch-bound and population-based algorithm are used to tackle the problem.

Since search dimensions of the combinatorial optimization in the mixed integer design increase exponentially with the number of targets, branch and pruning operations are frequently used to reduce the

☆ This paper is an extension of the paper originally presented as "Optimal Debris Removal Sequence with Multiple Spacecraft Using Non-Population Gradient Search" in the 73rd International Astronautical Congress, Paris, France, 18 - 22 September 2022.

* Corresponding author.
E-mail address: houliqiang@sjtu.edu.cn (L. Hou).

https://doi.org/10.1016/j.actaastro.2023.12.024
Received 17 June 2023; Received in revised form 16 November 2023; Accepted 13 December 2023
Available online 15 December 2023

searching subspaces and the computational cost. A notable example of tree searches in space mission design is the software STOUR developed by Jet Propulsion Laboratory. The automated design of trajectories with multiple fly-bys, designed with heuristic-based pruning [5], has been used in several important mission designs [6,7]. In [8], the population-based evolutionary algorithm was used to solve the static TSP. As for the dynamic TSP of the mission design, the bio-inspired methodology, such as Ant Colony Optimization (ACO) [9], Genetic Algorithm (GA) [10,11] is used to determine the optimal tour. In [12], a heuristic-free approach using Monte Carlo Tree Search (MCTS) is proposed for the automated encounter sequence planning. Problem-specific modifications are made to adapt traditional MCTS to trajectory planning of the Rosetta and Cassini Huygens interplanetary mission design.

To tackle efficiently the combinatorial optimization problem, adaptations and techniques are developed to the classic global optimization of the TSPs, e.g., Ant Colony Optimization [13], Genetic Algorithm [14,15], Two-Stage Dynamic-Assignment [16], Branch-and-Bound [17], and the Column-Generation technique [18], etc. In [19], a new global optimization algorithm, ACO with Timeline Club, is proposed for the time-dependent TSP of debris removal. Different to the classic ACO, a new structure, the Timeline Club, is used and replaces the pheromone matrix of the ACO. T.Uriot et al. consider machine learning in spacecraft collision avoidance, modeling and predicting the collision risk level between the spacecraft and orbiting objects over time. Competition of such prediction with machine learning using a large data set of close approach events with conjunction data messages (CDMs) is designed [20]. Design and results of the competition, including the training sets, etc., are described and presented.

Though several algorithms can be used to tackle the TSP space mission design, searching for the optimal path planning is computationally costly and difficult to use in many practical designs. Taking the population-based algorithm for example, to ensure diversity of the population, the population size of the algorithm is usually set to around 30–50 times the number of parameters to be optimized. In some practical space TSP designs, the number of candidate targets exceeds tens or even hundreds. Considering that the target and the continuous maneuvering parameters are optimized, the number of individuals in the algorithms could be up to thousands or tens of thousands depending on the mission situations. The computational cost can be tremendously huge because of the possible pathing branches. A high-performance computer is therefore required. As for the more complex scenarios of multiple objectives optimization of the space TSP, the situation could be worse.

From the analysis, it can be seen that there is little room left to improve the conventional TSP methods. The reason comes from the mixed integer essence of TSP. Given the mixed integer operations, the branches of path planning should be considered in whatever algorithms used. Since the separated sub-design spaces complicate the computation, if one can bridge the separated sub-spaces of TSP and map them into a continuous design space, the branching issues will be avoided, and a continuous optimizer can be used to determine the sequence instead. In particular, if a gradient-based optimizer can be used, the searching process will be significantly improved.

Note that the path planning shares similarities to sequential parameter estimation, given the fact that states of targets are varied and conditioned on preceding operations of the targets. We therefore introduce expected transfer trajectory, to measure not only deterministically but probabilities level of varied debris visits. Based on this idea, a new approach for the space TSP is developed in this work. In contrast to the conventional methods, most of which are designed based on branch-based operations, the new method models the mixed integer TSP as *a continuous parameter estimation problem*. Instead of the integer type ID, expected positions and associated variance of the nodes are set to the design variable. Uncertainty impacts due to determining the preceding target are considered. Then, the covariance and uncertainty noise of the preceding node is propagated and integrated into the estimation of the next node on the path. Finally, a probabilistic metric-based analysis is used to determine the target. With this analysis, one does not need to consider the separated subspace generated by the integer-type design variables. A gradient-based optimization is used directly to determine the optimal path. No branch and prune operation is implemented during the search.

With the method, a set of classic TSP benchmarks with different number of nodes are tested. A dynamic TSP of multiple debris rendezvous is then considered. Besides the targets on tour, the time of flight and maneuver parameters are also optimized. The sequential visit of near-earth debris considering J2 effects, with a single and multiple spacecraft are constructed.

The remainder of the paper is organized as follows. Section 2 formulates the TSP in view of Markov process, and models the TSP into a continuous parameter equation with uncertainties. The non-population gradient-based search method of static TSP is developed and presented in Section 3. Three classic TSP benchmarks are tested and compared to the actual optimal solutions. The method is then extended to dynamic TSP of multiple debris rendezvous in Section 4. Numerical simulations for the single and multiple spacecraft debris removal are presented in Section 5, and finally Section 6 concludes the paper.

2. Formulation of the TSP

2.1. Problem statement

Consider N nodes on a graph, $\mathbf{I} = \{\mathcal{I}_i\}_{i=1,\dots,N}$, where $\mathcal{I}_i \in \mathbb{Z}$ is the ID of the nodes. For each pair of different nodes, $\{\mathcal{I}_i, \mathcal{I}_j\}$, one has a distance as $d_{i,j}$. The goal of TSP is to find an ordering π of the nodes permutation and minimizes the total tour length

$$\min_{\pi} J = \sum_{i=1}^{N-1} d\left(\mathcal{I}_{\pi(i)}, \mathcal{I}_{\pi(i+1)}\right) \tag{1}$$

As for the dynamic TSP, an extra equation for the dynamics is considered, and the equations below can be used to express the problem

$$\min_{\pi} J = \sum_{i=1}^{N-1} y\left(\mathcal{I}_{\pi(i)}, \mathcal{I}_{\pi(i+1)}\right) \tag{2}$$

subject to

$$\dot{\mathbf{x}}_i = f(\mathrm{x}_i) \tag{3}$$

where f describes the dynamic equations, $\mathbf{x}_i = \mathbf{x}(\mathcal{I}_i)$ are the state of the $\mathcal{I}_i$, and $y\left(\mathcal{I}_i, \mathcal{I}_j\right)$ is the cost from $\mathcal{I}_i$ to $\mathcal{I}_j$ with the time of flight t_{ij}.

For simplicity, in the following sections, we use $\mathbf{x}_i \in \mathbb{R}^n$ and $y_i \in \mathbb{R}$ to represent the distance vector from $\mathcal{I}_{i-1}$ to $\mathcal{I}_i$ and the cost or edge distance between the two nodes. The goal of TSP is to determine an optimal sequence to minimize the total cost

$$\min_{\mathcal{I}_i \in \mathbf{I}} J = \sum_{i=1}^{N} y_i \tag{4}$$

where $\mathbf{I} = \{\mathcal{I}_j\}$ with $j = 1, \dots, N$ is the candidate set of the nodes to visit. In the problem, combinatorial operations are implemented. The branch and pruning operations are therefore commonly used in graph-related TSP. In this work, we try to model the problem in a different way, model it into a serial mode, and tackle it using a gradient-based estimator.

Note that in the classical TSP, the agent returns to the site at which he/she initiates the route. In this work of path planning of debris removal, the spacecraft does not need to return to the original site as in the classical TSP. However, one can still use the relevant equations for path planning for debris removal.

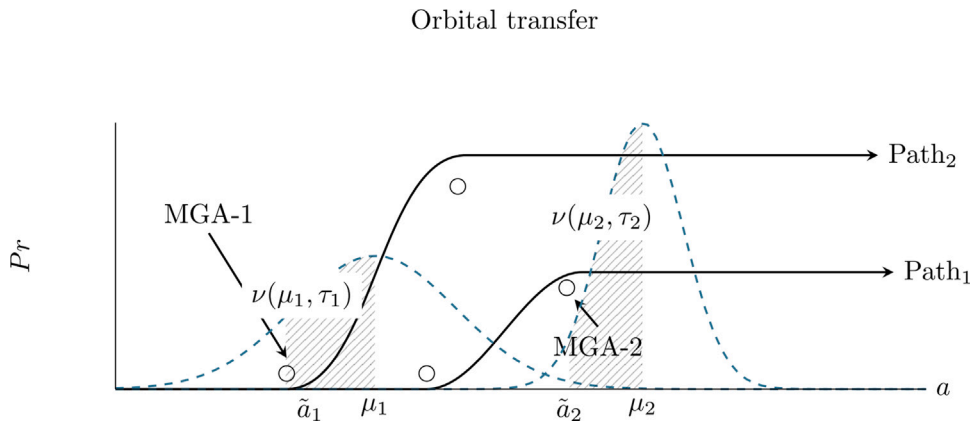

Fig. 1. Similarity metric of multiple gravity assist (MGA) [21].

2.2. Stochastic essence of TSP

From the definition of the TSP, it can be seen that for each node in the list of TSP targets, a position vector is given uniquely to a target's identification number. The position vector can uniquely express the node instead of the integer ID to be determined. One can use the expected position of the node as the design vector to optimize the mixed integer TSP. However, given the expected position value as a design variable, there are still some problems in determining the sequence. For example, suppose part of the nodes on the path has been known. Given the expected position and a list of candidate targets, a candidate target or ID should be selected as the node to visit next based on the accumulated information of the preceding nodes. A metric should therefore be defined to show how the expected position is close to the actual one. The metric should consider not only the distance of candidates to the current node, but the impact of the preceding nodes. Propagation of uncertainty during decision-making and the likelihood of the planning should be considered as well.

For its stochastic essence, the GA and its variations can be used for such a TSP, for example, the multi-gravity assist design. The individual chromosome of GA is encoded with the integer ID and continuous maneuver parameters. The cost of each TSP path is computed and compared. Individuals with better performances are selected to re-generate new populations using genetic operations, such as mutation and crossover. In [21], a likelihood analysis-assisted evolution for the GA is further developed to optimize the multiple gravity assist. The expected position and variances of the planet instead of the ID are set to the design variables. The covariance acts as the uncertain metric of the parameter estimator, and is used to exclude the branches having low-level likelihood.

Fig. 1 shows the flying paths and bandwidths for filtering the low-level branches. The abscissa represents semi-major axis, a characteristic parameter of the orbital transfer trajectory which can be seen as the energy level of the orbit. The y-axis (Pr) represents the candidate planet's probability of flying by. The solid line and dashed line schematically represent the optimal and candidate paths, respectively. The circles represent planets. Similarity metric measure of the planetary orbit to the expected spacecraft trajectory, i.e., the shaded area is defined by bandwidth τ, and the difference between semi-major of the planet $\tilde{a}$ and predefined expected semi-major axis $\mu.\tau$ acts as the filter bandwidth to filter out candidates that cannot meet the swing-by requirements. The greater the metric value is, the higher probability that the planet could be selected as swing-by planet. Both μ and τ are set as tuning parameters in the optimization to maximize likelihood of the objective function value. A subroutine is designed and implemented for statistically analyzing the impact on the transfer cost due to uncertainty concerning the varied expected position. The new GA algorithm is implemented in continuous space and outperforms the conventional GA. As a result, the risk of missing valuable mutation candidates in the mixed integer space is reduced.

2.3. TSP and Markov process: The system equations

Though the likelihood analysis significantly improves the classical mixed integer GA, the branching operation is implemented during the searches of the GA. Since the path and cost of TSP are expressed sequentially, it would be better to express and implement the optimization in sequential mode too. We therefore introduce the metric for measuring the uncertainty of the expected value of the target with respect to the actual node. The uncertainty during the path planning should consider not only the uncertainty of the current position, but also the impact related to the preceding node. This leads to the interesting comparison of the TSP to the Markov chain. By definition, the Markov chain is a stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event.

Compare the process to the TSP path determination. The possible path to be explored are expressed using the expected path associated with the probability. A Gaussian model is assumed and used in the uncertainty analysis. Covariance with the distance vector to the node is used to measure the probability and uncertainty in decision-making. The following equation can be used to express the relation of the position of $\mathcal{I}_i$ conditioned on $\mathcal{I}_{i-1}$

$$\hat{\mathbf{x}}_{i|i-1} \sim \mathcal{N}\left(\bar{\mathbf{x}}_{i|i-1}, \mathbf{\Sigma}_{i|i-1}\right) \tag{5}$$

where $\bar{\mathbf{x}}_{i|i-1}$ and $\mathbf{\Sigma}_{i|i-1}$ are the expected value and covariance of the ith node conditioned on the $i-1$-th node.

As for the actual value of the ith node, the actual position can be modeled as

$$\hat{\mathbf{x}}_i = \bar{\mathbf{x}}_{i|i-1} + \boldsymbol{\sigma}_{i-1} + \mathbf{v} \tag{6}$$

where $\boldsymbol{\sigma}_{i-1} \sim \mathcal{N}\left(0, \mathbf{\Sigma}_{i-1}\right)$ is uncertainty due to the selection of $\mathcal{I}_{i-1}$ and propagated to the ith node. $\mathbf{v} \sim \mathcal{N}(0, \mathbf{V})$ is the noise of the system equation. Corresponding transfer cost of the i-the segment, considering the decision impact, will be

$$\hat{y}_i = \bar{y}_{i|i-1} + \sigma_{y,i-1} + w \tag{7}$$

where $\bar{y}_{i|i-1}$ is the expected value of the cost of the ith segment conditioned on $\mathcal{I}_{i-1}$, $\sigma_{y,i-1} \sim \mathcal{N}(0, \Sigma_{y,i-1})$ is uncertainty of cost due to determination of $\mathcal{I}_{i-1}$ and propagated to the ith node, and $w \sim \mathcal{N}(0, W)$ is the noise in the cost equation.

The system and cost equation of TSP are similar to those of parameter estimation, except that the values to be estimated, $\bar{\mathbf{x}}_{i|i-1}$, $\Sigma_{i|i-1}$ are conditioned on the preceding node, while in the parameter estimation, the current state and covariance, $\bar{\mathbf{x}}_i$ and Σ_i are estimated. The model bridges the mixed integer TSP with the parameter estimation, which could potentially be used to tackle more complex graph-like scenarios. More detailed analysis and optimization of the estimates and parameters are presented in the following sections.

3. Static TSP and benchmark test

Consider a static TSP benchmark of TPSLIB [22], kroA100. The benchmark has 100 nodes to visit. The objective of the benchmark is to find an optimal travel route such that each point is visited once and only once with the shortest possible distance. With the system equations using the serial mode parameter estimate, we first present the prior of the TSP, analyze its impact and likelihood of decision making on the node, update the posterior of the decision, and then construct a sequence using maximum likelihood to determine. The optimal sequence is found using a gradient-based quadratic optimizer.

3.1. Prior of the TSP

Given a specific path $\{\mathcal{I}_i\}$, the nodes on the path can be characterized by distance vectors of the edges. Suppose the distance vector of $\mathcal{I}_i$ with respect to $\mathcal{I}_{i-1}$ is $\mathbf{x}_{i|i-1}$, and its covariance conditioned on $\mathcal{I}_{i-1}$ is $\boldsymbol{\Sigma}_{i|i-1}$. From the system equations, the distance vector and associated covariance can be used as metrics to determine $\mathcal{I}_i$. With this assumption, given the nodes on the path, $\mathcal{I}_1, \dots, \mathcal{I}_{i-1}$, the node $\mathcal{I}_i$ should minimize the cost of edge, y_i, while at the same time maximizing its likelihood. Priors and parameters to be estimated are summarized as follows:

(1) Prior of $\mathcal{I}_{i-1}$:

- $\hat{\mathbf{x}}_{i-1}$, estimated vector of $\mathcal{I}_{i-1}$;
- $\boldsymbol{\Sigma}_{i-1}$, covariance of $\mathcal{I}_{i-1}$.

(2) Parameters of $\mathcal{I}_i$ to be estimated:

- $\bar{\mathbf{x}}_{i|i-1}$, the expected value of the ith node's distance vector conditioned on $\mathcal{I}_{i-1}$. $\bar{\mathbf{x}}_{i|i-1}$ will be optimized in determining the sequence;
- conditional covariance $\boldsymbol{\Sigma}_{i|i-1} = \text{diag}\left(\sigma_{i|i-1}\right)$, where $\sigma_{i|i-1}$ is the variance of $\mathcal{I}_i$'s distance vector conditioned on $\mathcal{I}_{i-1}$, and will be optimized.

(3) Candidate set of the ith node:

- $\mathcal{I}_j \in \mathbf{I} \backslash \{\mathcal{I}_1, \dots, \mathcal{I}_{i-1}\}$, the candidate node;
- $\mathbf{x}_j$, distance vector of $\mathcal{I}_j$ with respect to $\mathcal{I}_{i-1}$.

3.2. Model the uncertainty impacts of the node's position

Impacts of uncertainty due to the preceding should be considered in determining nodes on the TSP path. The Gaussian distribution is used to model the noise of the decision uncertainty. The corresponding uncertainty of the cost is supposed to be Gaussian too. In the preceding section, we showed the relation between the estimate of $\mathcal{I}_i$ and the estimate of $\mathcal{I}_i$ conditioned on $\mathcal{I}_{i-1}$:

$$\begin{aligned} \hat{\mathbf{x}}_i &= \hat{\mathbf{x}}_{i|i-1} + \boldsymbol{\sigma}_{i-1} + \mathbf{v} \\ \hat{\mathbf{x}}_{i|i-1} &\sim N\left(\bar{\mathbf{x}}_{i|i-1}, \boldsymbol{\Sigma}_{i|i-1}\right) \end{aligned} \tag{8}$$

where $\mathbf{v} \sim \mathcal{N}(0, \mathbf{V})$ is the noise of the system equation, taking into account the uncertainty due to the distance metric of the selection. $\boldsymbol{\sigma}_{i-1} \sim N\left(0, \boldsymbol{\Sigma}_{i-1}\right)$ is the uncertainty due to selection of $\mathcal{I}_{i-1}$.

Since the TSP is static, the noise of decision propagated to the ith node has the same covariance as $\mathcal{I}_{i-1}$. With this assumption, expected value and covariance of estimated $\mathcal{I}_i$ can be obtained as

$$\bar{\mathbf{x}}_i = \bar{\mathbf{x}}_{i|-1}, \quad \boldsymbol{\Sigma}_{i|i} = \boldsymbol{\Sigma}_{i|i-1} + \boldsymbol{\Sigma}_{i-1} \tag{9}$$

Regarding the cost function, given $\mathcal{I}_{i-1}$, the estimate of the edge cost of $\mathcal{I}_i$ can be written as

$$\hat{y}_i = f\left(\hat{\mathbf{x}}_i\right) + w \tag{10}$$

where $w \sim N(0, \mathbf{W})$ is the uncertainty impact of the distance metric used in the cost function.

The expected value of the cost from $\mathcal{I}_{i-1}$ to $\mathcal{I}_i$ is computed as

$$\bar{y}_i = f\left(\bar{\mathbf{x}}_i\right) \tag{11}$$

with the variance

$$\sigma_{y,i} = \mathbf{F}_i \boldsymbol{\Sigma}_{i|i} \mathbf{F}_i^T \tag{12}$$

where $\mathbf{F}_i$ is the Jacobian of the cost y_i with respect to $\bar{\mathbf{x}}_i$

$$\mathbf{F}_i = \frac{\partial y_i}{\partial \bar{\mathbf{x}}_i} = \left[\frac{\partial y_i}{\partial \bar{x}_{i,1}} \cdots \frac{\partial y_i}{\partial \bar{x}_{i,n}}\right] \tag{13}$$

The relations given in Eqs. (9)–(13) are critical to the TSP estimation. By using these relations, one can construct an iterative process for determining TSP sequences. *Continuous* probability metric is used instead to prune the unnecessary branches. Also, population-based evolutionary operations, such as genetic procedures and ant colony algorithms, can be avoided.

3.3. Determine the node

Suppose at the ith node, the candidate set of nodes are $\{\mathcal{I}_j\}$. The candidates are supposed normally distributed with respect to expected $\mathcal{I}_i$, and have

$$\mathbf{x}_j \sim \mathcal{N}\left(\bar{\mathbf{x}}_i, \boldsymbol{\Sigma}_{i|i}\right) \tag{14}$$

where $\mathbf{x}_j$ represents the relative distance of $\mathcal{I}_j$ to $\mathcal{I}_i$, and $\bar{\mathbf{x}}_i$ represents the expected distance of $\mathcal{I}_i$. Note that covariance $\Sigma_{i|i}$ is obtained considering not only the covariance conditioned on $\mathcal{I}_{i-1}$, but the covariance of $\mathcal{I}_{i-1}$ propagated to $\mathcal{I}_i$ (Eq. (9)).

Given expected position $\bar{\mathbf{x}}_i$, the candidate to be selected should have the maximum logarithm likelihood. Therefore, the candidate is selected according to the following equation:

$$\begin{aligned} \min_j \mathcal{L}_j = \min_j & \frac{1}{2}\left(\mathbf{x}_j - \bar{\mathbf{x}}_i\right)\left(\boldsymbol{\Sigma}_{i|i} + \mathbf{V}\right)^{-1}\left(\mathbf{x}_j - \bar{\mathbf{x}}_i\right)' \\ & + \frac{1}{2}\log\left(\det \boldsymbol{\Sigma}_{i|i} + \mathbf{V}\right) \end{aligned} \tag{15}$$

where $\mathbf{V} = \text{diag}\left(v_i\right)$ represents the covariance matrix of the noise caused by the distance metric used in the system equation.

3.4. Update covariance

If $\mathcal{I}_i$ is estimated to be $\hat{\mathbf{x}}_i$ with the cost $\hat{y}_i$, a posterior estimate and covariance can be calculated using the new determined $\mathcal{I}_i$. Suppose estimates of the cost $\hat{y}_i$ and the states $\hat{\mathbf{x}}_i$ are jointly Gaussian, i.e.,

$$\begin{bmatrix} \hat{\mathbf{x}}_i \\ \hat{y}_i \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \bar{\mathbf{x}}_i \\ \mathbf{F}_i \bar{\mathbf{x}}_i \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{i|i} & \boldsymbol{\Sigma}_{i|i}\mathbf{F}_i^T \\ \mathbf{F}_i \boldsymbol{\Sigma}_{i|i} & \mathbf{F}_i \boldsymbol{\Sigma}_{i|i}\mathbf{F}_i^T + W \end{bmatrix}\right) \tag{16}$$

where $\mathbf{W}$ is covariance of the cost function. Variations of the cost less than the noise cannot be considered in determining the sequence. Posterior covariance of $\hat{\mathbf{x}}_i$ can be obtained as

$$\boldsymbol{\Sigma}_i = \boldsymbol{\Sigma}_{i|i} - \boldsymbol{\Sigma}_{i|i}\mathbf{F}_i^T\left(\mathbf{F}_i \boldsymbol{\Sigma}_{i|i}\mathbf{F}_i^T + \mathbf{W}\right)^{-1}\mathbf{F}_i \boldsymbol{\Sigma}_{i|i} \tag{17}$$

In Eq. (17), $\boldsymbol{\Sigma}_{i|i}$ is the estimate of covariance of $\mathcal{I}_i$, and obtained using Eq. (9), while $\boldsymbol{\Sigma}_i$ is the posterior covariance of $\mathcal{I}_i$.

Propagating the covariance to $\mathcal{I}_{i+1}$ to update the estimate of covariance, one has

$$\boldsymbol{\Sigma}_{i+1|i+1} = \boldsymbol{\Sigma}_{i+1|i} + \boldsymbol{\Sigma}_i \tag{18}$$

New covariance $\boldsymbol{\Sigma}_{i+1|i+1}$ can now be used to estimate the next node.

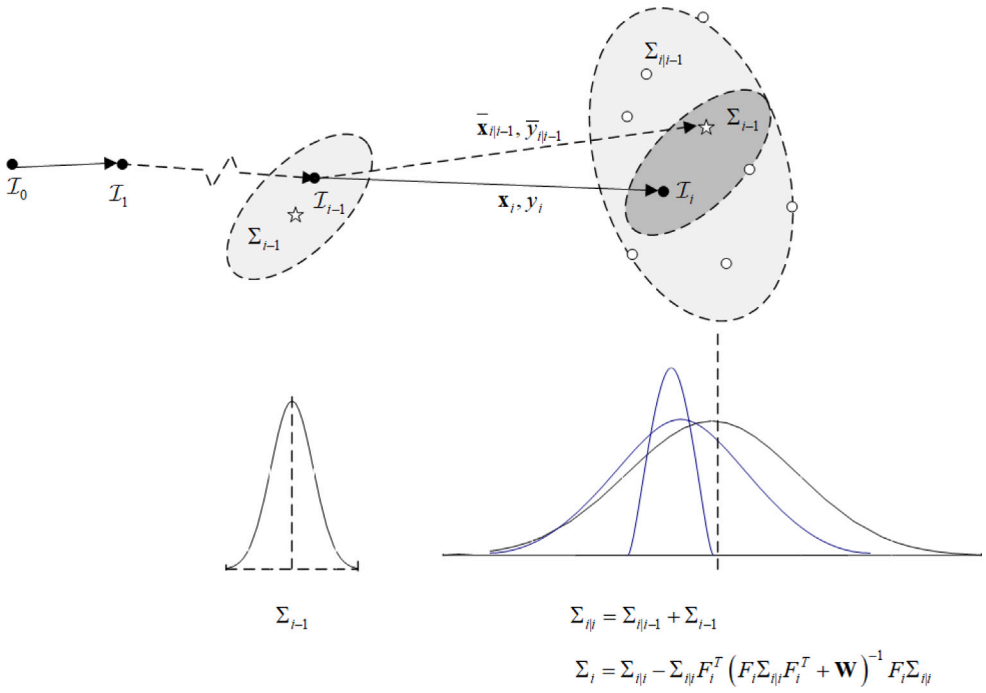

Fig. 2. Uncertainty propagation of TSP. Hollow circle represents the candidate node, while the solid circle and the star represent the selected node and expected node, respectively.

Fig. 2 schematically illustrates TSP path uncertainty propagation. The hollow circle represents the candidate node, while the solid circle and the star represent the selected node and expected node, respectively. Uncertainty of the node $\mathcal{I}_{i-1}$, Σ_{i-1} is propagated and considered in determining the node of $\mathcal{I}_i$. The uncertainty after selection, the posterior, Σ_i is obtained and improved with $\Sigma_{i|i}$. Grid size of the selection, or the noise during the decision making is considered and analyzed in the section of noise of parameter settings of the noise.

3.5. Parameter setting of the noise

In the system and cost equations, Eqs. (8) and (10), the noise $\mathbf{v} \sim \mathcal{N}(0, \mathbf{V})$ and $w \sim N(0, \mathbf{W})$ are introduced and considered. The noise is necessary in path planning of the sequence. Suppose in the TSP distance couples database, the minimum value of design space $\mathbf{X}$ and objective space $\mathbf{y}$ is $\mathbf{x}_{min}$ and d_{min}, respectively. The decision for the current node will keep unchanged if the design variable $\mathbf{x}_i$ and the distance value y_i are less than $\mathbf{x}_{min}$ and d_{min}. Consideration should be given to the impact of such operations when planning nodes on the path.

Such operations act as noise or grid size in decision-making. A noise setting is, therefore, necessary and should be considered, and a non-zero noise value must be given to model its impacts. The value is set to $\mathbf{V} = \text{diag}\left(\upsilon_i\right) \in \mathbb{R}^{n \times n}$ for the design space, and $\mathbf{W} \in \mathbb{R}$ for the cost equation respectively. Element of the noise, υ_i, is set to equal to the minimum edge length of each dimension, i.e., $\upsilon_1 = \min_i \left|x_{i,1}\right|, \ldots, \upsilon_n = \min_i \left|x_{i,n}\right|$; variation of the edge length less than υ_i in the candidate selection will be neglected. The noise $\mathbf{W} = \min_{i,j} \left|d_{i,j}\right|$ is set to the minimum cost of the edge between the nodes, where $d_{i,j} \in \mathbb{R}$ is cost between $\mathcal{I}_i$ and $\mathcal{I}_j$. Variations of the cost that are less than the noise will be neglected and cannot be considered valid in determining the sequence.

3.6. Construct the sequence

Eq. (15) shows the criterion conducted in design space. For each individual node, the candidate of maximum likelihood with respect to the expected target is selected. However, the total cost of the visit is not necessarily optimal with these operations, since the optimal sequence should consider not only the criteria in design space but also the cost equation. In addition to the operation of likelihood-based selection for the target $\mathcal{I}_i$, the accumulated likelihood of the segment cost y_i should be considered too.

Considering the cost equations, Eqs. (11)–(12), given the node $\mathcal{I}_i$, logarithm likelihood of its cost y_i can be computed as

$$\begin{aligned}\mathcal{L}_i = &\frac{1}{2}\left(y_i - \bar{y}_i\right)\left(\mathbf{F}_i \boldsymbol{\Sigma}_{i|i} \mathbf{F}_i' + \mathbf{W}\right)^{-1}\left(y_i - \bar{y}_i\right)' \\ &+ \frac{1}{2}\log\det\left(\mathbf{F}_i \boldsymbol{\Sigma}_{i|i} \mathbf{F}_i' + \mathbf{W}\right)\end{aligned} \tag{19}$$

where $y_i = f(\mathbf{x}(\mathcal{I}_i))$ is the cost to the node $\mathcal{I}_i$, $\mathbf{W} \in \mathbb{R}$ is the covariance and noise of the cost function.

The prior for constructing the sequence is $\theta_i = \left[\bar{\mathbf{x}}_{i|i-1}, \boldsymbol{\sigma}_{i|i-1}\right]$. The following equation can then be used to optimize the whole path

$$\min_{\theta_i} J = \sum_{i=1}^{N} y_i + \sum_{i=1}^{N} \mathcal{L}_i\left(\theta_i\right) \tag{20}$$

A continuous optimizer can be used to search the optimal solution $\theta_i^* = \left[\mathbf{x}_{i|i-1}^*, \boldsymbol{\sigma}_{i|i-1}^*\right]$. As $\mathcal{L}_i\left(\theta_i\right)$ is quadratic concave, a gradient-based optimizer, such as Sequential Quadratic Programming (SQP), can be used given that the priors are in a certain range neighbor to the optima. Fig. 3 shows the flowchart of the proposed algorithm.

3.7. TSP and EKF parameter estimation

Table 1 compares TSP and parameter estimator. An Extended Kalman Filter (EKF)-like parameter estimator is listed and compared. In both, the system equation and cost equation are similar; both are optimized through likelihood analysis and are designed according to Markov chains. However, in the TSP design, the conditioned state and variance $\bar{\mathbf{x}}_{i|i-1}$ and $\Sigma_{i|i-1}$ are optimized, whereas in the classic parameter estimator, the current state $\bar{\mathbf{x}}_i$ and its covariance Σ_i are estimated based on the observations y_i. The posterior equations are similar too. However, the estimated covariance of TSP $\Sigma_{i|i}$, considers the uncertainty from the preceding node, whereas in the parameter estimator, only the current covariance Σ_i is used to improve the posterior.

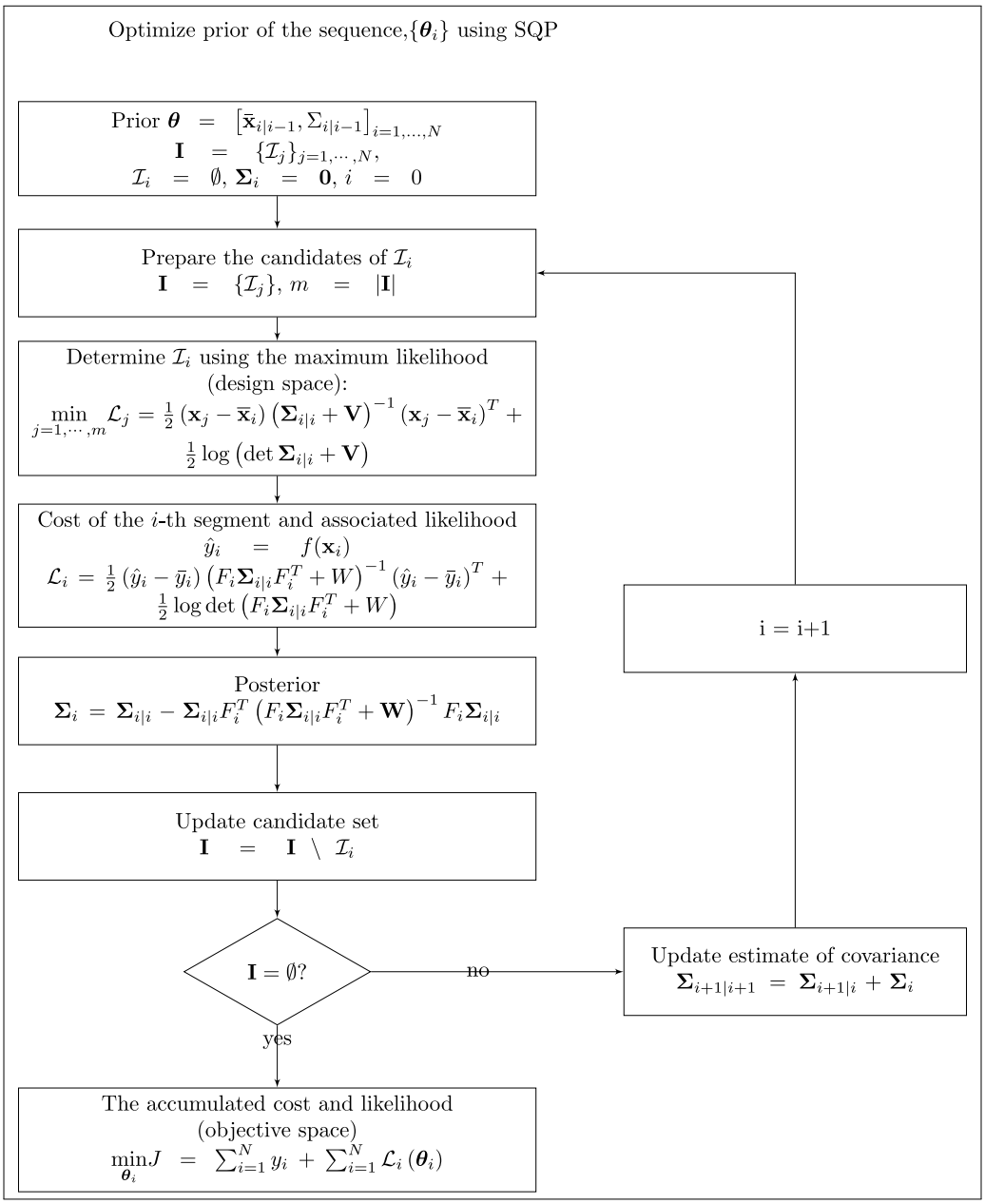

Fig. 3. Flow chart of the TSP node determination.

In this work, a gradient optimizer is used for the TSP, while in the EKF, such operations do not exist. The parameter estimation converges iteratively to the optimal estimates given a series of observations. Though the gradient based optimizer, we hope that in the future we can find another way as opposed processing a large scale space TSP. In the mega-constellation, the number of satellites can be up to thousands.

In the static TSP, propagation of uncertainty impacts of the preceding node is straightforward, and equals to the posterior of the preceding node. In the dynamic TSP, however, the dynamics should be considered when evaluating uncertainty impacts. We will discuss this in the following sections.

3.8. Test of benchmark

A classic TSP benchmark, kroA100, is used to test the proposed method. The objective of the benchmark is to find a round trip of minimal total length visiting each node exactly once. The benchmark is from TSPLIB, a library of sample instances for the TSP from various sources and of different types [22]. The benchmark has 100 nodes. Coordinates of the nodes and the distances are from the database at TSPLIB. In the test of the benchmark, the Euclidean distance is used. Jacobian of the distance with respect to $\mathbf{x}_i = \left[x_i, y_i\right]$ in Eq. (13) can therefore be explicitly given and computed as

$$\mathbf{F}_i = \left[\frac{x_i}{\sqrt{x_i^2 + y_i^2}}, \frac{y_i}{\sqrt{x_i^2 + y_i^2}}\right] \tag{21}$$

Table 2 shows the parameter settings of the TSP algorithm. Design parameters of the algorithm consist of the expected value and the conditional variance of distance vector, $\boldsymbol{\theta}_i = \left[\bar{\mathbf{x}}_{i|i-1}, \boldsymbol{\sigma}_{i|i-1}\right]$. The design vector is scaled into $[0, 1]$ with the minimum and maximum edge distance. Initial value of $\bar{\mathbf{x}}_{i|i-1}$ is set to the minimum edge distance in each dimension, $\left[|x|_{\min}, |y|_{\min}\right]$, and scaled into $[0, 1]$.

Similar to continuous parameter estimators, such as Kalman filter, in the test of continuous TSP solver, a proper setting of the noise of

Table 1
TSP vs. Parameter estimator.

	TSP	Parameter estimator
System equation	$\hat{\mathbf{x}}_i = \hat{\mathbf{x}}_{i\|i-1} + \boldsymbol{\sigma}_{i-1} + \mathbf{v}$	$\hat{\mathbf{x}}_i = \hat{\mathbf{x}}_i + \mathbf{v}$
Observation	$\hat{y}_i = f\left(\hat{\mathbf{x}}_{i\|i-1}\right) + w$	$\hat{y}_i = f\left(\hat{\mathbf{x}}_i\right) + w$
Variables to be estimated	$\mathbf{x}_{i\|i-1},\ \boldsymbol{\Sigma}_{i\|i-1}$	$\mathbf{x}_i,\ \boldsymbol{\Sigma}_i$
Update of the covariance	$\boldsymbol{\Sigma}_{i\|i} - \boldsymbol{\Sigma}_{i\|i}\mathbf{F}_i^T\left(\mathbf{F}_i\boldsymbol{\Sigma}_{i\|i}\mathbf{F}_i^T + \mathbf{W}\right)^{-1}\mathbf{F}_i\boldsymbol{\Sigma}_{i\|i}$ $\boldsymbol{\Sigma}_{i+1\|i+1} = \boldsymbol{\Sigma}_{i+1\|i} + \boldsymbol{\Sigma}_i$	$\boldsymbol{\Sigma}_i - \boldsymbol{\Sigma}_i\mathbf{F}_i^T\left(\mathbf{F}_i\boldsymbol{\Sigma}_i\mathbf{F}_i^T + \mathbf{W}\right)^{-1}\mathbf{F}_i\boldsymbol{\Sigma}_i$

Table 2
Parameter setting of test of benchmark kroA100.

Parameter	Lower bound	Upper bound	Initial value
$\bar{x}_i$	0.0	1.0	$0.5 - \frac{\lvert x\rvert_{\min}}{\lvert x\rvert_{\max} - \lvert x\rvert_{\min}}$
$\bar{y}_i$	0.0	1.0	$0.5 - \frac{\lvert y\rvert_{\min}}{\lvert y\rvert_{\max} - \lvert y\rvert_{\min}}$
$\sigma_{x,i}$	0.0	1.0	1.0
$\sigma_{y,i}$	0.0	1.0	1.0

Table 3
Noise matrix of the test.

Benchmark	Noise of system (V)	Noise of cost (W)
kroA100	$\mathrm{diag}\left(\left[1.0^2, 1.0^2\right]\right)$	13.038^2

Table 4
Sub-paths vs. optimal sub-paths.

Sub-sequence (cost)	Optimal sub-sequence (cost)	Number of nodes	Error (percentage)
90-79-53-19⋯-94-18 (5262.3)	90-19-75-⋯-79-18 (5174.1)	23	1.7%

the system and cost equation is required. The noise matrix, or the grid size of the optimization in Eqs. (15) and (17), V and **W**, is set equal to the square of the minimum edge distance to take into account the uncertainty of the distance metric. Table 3 shows the parameter setting of the noise of the system equation and cost equation.

The first and ending nodes of the path are set to 1. A Matlab sqp optimizer, fmincon, is used to search the optimal path. Fig. 4 shows the benchmark's optimal sequence and tour cost, respectively. The benchmark results are listed and compared to the actual optimal solutions. The relative deviation of the tour cost after around 50 iterations is within 9.8%. The results given by the algorithm are generally close to the optimal tour path. The source code of test of the benchmarks can be available at: https://github.com/LiqiangHou/Travelling-Salesman-Problem-of-Space-Trajectory-Design.

In the source code we list some other experimental results of the TSP with 17 and 37 nodes as well. The results are compared to those obtained using simulated annealing and genetic algorithm. These two are popular methods for the TSP-like problem. The results look almost the same, within a range of less than 10%. Figs. 5 and 6 compare the results to the optimal solutions.

With the results shown in Fig. 4, it can be seen that the path generally capture the optimal path, even some edges with the minimum distance, 31–80, 5–37, 92–28, and 13–76, etc. can be determined. However, in some sub-sequences in the figure, the path is crossed and trapped in the local optima. Table 4 lists and compares one of the sub-sequences to the actual optimal ones.

In the sub-sequence, given the node $\mathcal{I}_{i-1} = 90$, expected value and covariance of the sub-sequence are

$$\bar{\mathbf{x}}_{i|i-1} = [-21.756, 6.4101] \text{ and } \boldsymbol{\sigma}_{i|i=1} = \begin{bmatrix} 8301.4 & -2108.7 \\ -2108.7 & 3364.4 \end{bmatrix}$$

respectively. Table 5 lists part of the candidate nodes of $\mathcal{I}_i$, $\{C_1, \ldots, C_4\}$. The nodes are sorted using the likelihood with respect to $\bar{\mathbf{x}}_{x|i-1}$. The node with maximum likelihood is C_1, with the edge distance 364.72. Note that in the table, edge cost of the node C_2, is less than C_1.

Table 5
Candidate nodes and likelihood given the node $\mathcal{I}_{i-1} = 90$.

Candidate	ID	Negative likelihood	x	y
C_1	79	32.83	[−364, 23]	364.72
C_2	10	48.42	[−4, 300]	300.02
C_3	84	55.71	[−131, 359]	382.15
C_4(∗)	19	71.12	[−131, −352]	375.58

The algorithm successfully avoids the local optima C_2 by using the likelihood metric, but failed to find the actual optimal node C_4 marked with star in the table. The reason might be related to the cost metric and the design variables of the algorithm.

The value of $\bar{\mathbf{x}}_{i|i-1}$ and $\boldsymbol{\sigma}_{i|i-1}$ are determined by the SQP optimizer. In the objective function of whole path, the accumulated likelihood of edge distances is quadratic (Eq. (20)), however, the sum of distances itself is not quadratic. The optimality of searching is therefore not necessarily global. Another problem might come from the design variables. To simplify the searching process, at the current stage, impacts between the components of $\mathbf{x}$, x_1 and x_2 are not included. Axis of the error ellipsoid are supposed to be identical to x_1 and x_2. Performances of the searching might be able to be improved if such impacts are included in the computation. More sophisticated cost metric and impact analysis techniques will be considered in the future studies.

3.9. Discussions

We note that for an optimal path, given an expected value of such a path, one needs to compare the path not only by measuring the variables in the design space but the cost should also be considered. Therefore, in the algorithm, we designed two criteria in both the design and objective space to ensure that the path is as close as possible to the optimal path. First, a design space criterion using the likelihood is expressed as Eq. (15). The equation selects the node closest to the expected one. As for the cost, the total cost, together with the accumulated likelihood, is computed with Eqs. (19) and (20).

In this method, we use the normal distribution-based metric for the priority level measurement. The exponential-based metric helps distinguish the candidate from those nearby, while the simple Euclidean distance cannot work as effectively when the candidates are clustered. The logarithms are quadratic and can be easily searched for its concave characteristic. It can be seen in the test that because of the introduced logarithm likelihood criteria, the concavity of both the design space and the cost function is improved, and the near-optimal solutions are obtained.

In the classic TSP, the cost function is continuous and differentiable with respect to the node position. While for some complex TSP-like problems, the cost function can be discrete and non-differentiable, e.g. a TSP for visiting the targets with different level merit values. In this case, if a new exponential form distance metric can be defined, the logarithm of the metric can be used as likelihood. After this, the proposed method might be able to be used to optimize the path. That would provide potential flexibility to some space TSP problems that current optimization methods cannot tackle.

Last but not least, the method can be easily extended to dynamic TSP. The covariance is used in the new TSP algorithm. The expression can be easily extended to the dynamics TSP case since the covariance

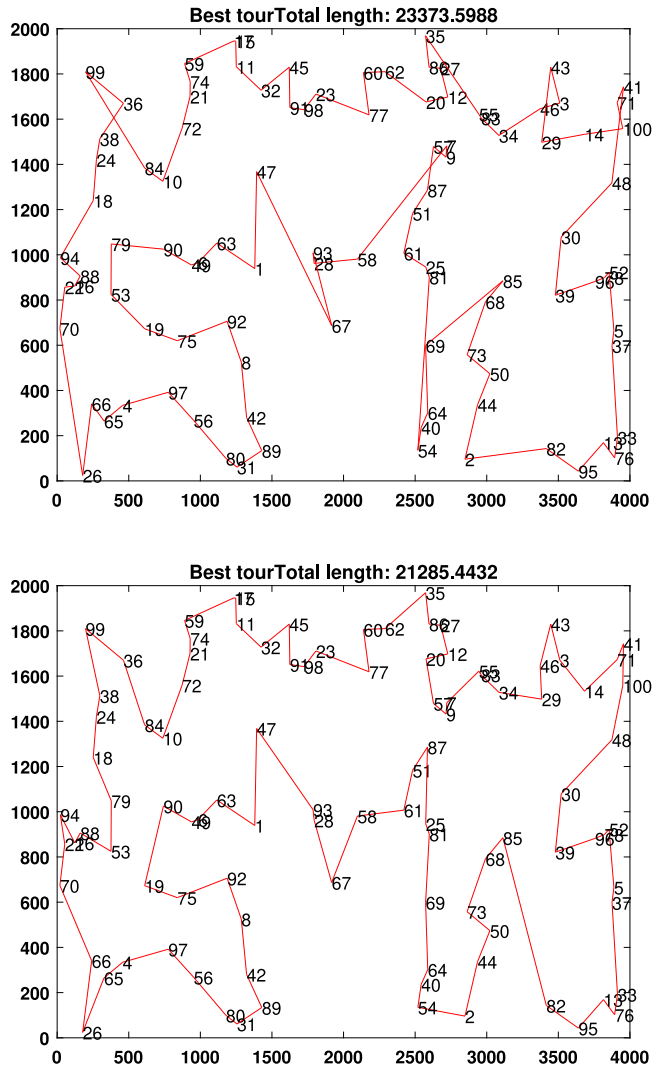

Fig. 4. Benchmark kroA100: Test results of the proposed TSP algorithm (top) vs. the optimal solution(bottom).

can be obtained and propagated through the dynamic equation. In contrast, for the conventional methods with geometric metrics and branching operations, it could be hard to use the static TSP technique directly to the dynamic ones. We will see in the following sections that only one extra step is required in the space TSP with the new algorithm.

With the conventional method of space TSP, a predefined grid set in the time domain is usually given. Then, trajectories between the nodes are computed with the Time of Flight and dynamics. The data and the grid size will significantly influence the search process. Some researchers, therefore, proposed the surrogate-based method to reduce the computational cost. Many computations are required before extending the static TSP operations to the space dynamic TSP. In contrast, such sampling operations can be avoided with the proposed method. Characteristic parameters of the distribution given Time of Flight can be predictable during the path construction.

In some tests of TSP, the number of instances can be huge; e.g., TSP for the 13509 U.S. cities with a population over 500 in an American city route. This study shows an algorithm for the medium-size TSP with about 100 targets. In our test of space mission design, the number of targets is also about 100. Further techniques for tackling large-scale space TSP design, such as mega-constellations, will be considered in future works.

4. Dynamic TSP of multiple debris rendezvous

The debris removal sequence problem is designed for the Sun-synchronous, Low-Earth-Orbit debris. The problem is essentially an optimization of path planning too, considering the dynamics involved. We first present the dynamics associated with the orbital service. A TSP like formulation for the multiple debris removal is developed and compared to the static TSP. An optimization considering the dynamics for the TSP is then constructed and tested.

4.1. Multiple debris rendezvous dynamics

4.1.1. Equations of motion

Suppose the debris are in the Sun-synchronous, Low-Earth-Orbits. Precession rates of the debris considering J2 coefficient, given the orbital element $[a, e, i, \Omega, \omega, M]$, can be computed using

$$\dot{\Omega} = -\frac{3}{2} J_2 \left(\frac{r_{eq}}{p}\right)^2 n \cos i \tag{22}$$

$$\dot{\omega} = \frac{3}{4} J_2 \left(\frac{r_{eq}}{p}\right)^2 n \left(5\cos^2 i - 1\right) \tag{23}$$

where r_{eq} is the mean radius of the Earth's equator, and $n = \sqrt{\frac{\mu}{a^3}}$ is the mean motion. The inclination of the orbit is i, and p is the semilatus rectum and computed as $p = a\left(1 - e^2\right)$. Corresponding right ascension of the ascending node, Ω, argument of perigee, ω, and the mean anomaly M at epoch t are given by

$$\Omega - \Omega_0 = \dot{\Omega}\left(t - t_0\right) \tag{24}$$

$$\omega - \omega_0 = \dot{\omega}\left(t - t_0\right) \tag{25}$$

$$M - M_0 = n\left(t - t_0\right) \tag{26}$$

Initial values of the debris ephemerides, $\left[a_0, e_0, i_0, \Omega_0, \omega_0, M_0\right]$ and initial epoch t_0 are taken from the GTOC-9 database.

4.1.2. Cost of near-circular impulsive orbital transfer

Based on the equation of motion with J2, different techniques of analytical approximations for the velocity increment computation of rendezvous are developed [23–25]. In the following, an approximation of velocity cost for node drift $\Delta\Omega$ and node drift rate, $\Delta\dot{\Omega}$ is developed. Cost for the circular orbital transfer rendezvous can be obtained via analytical computations.

Suppose the initial orbital element of the spacecraft at time t_0 is $\boldsymbol{\sigma}_0 = \left[a_0, e_0, i_0, \Omega_f, \omega_0, M_0\right]$. Then the ΔV for spacecraft to fly and rendezvous with the target with orbital element vector $\boldsymbol{\sigma}_f = \left[a_f, e_f, i_f, \Omega_f, \omega_f, M_f\right]$ after time of flight Δt can be obtained using Edelbaum's approximation [26]

$$\Delta V_a = \frac{1}{2}\frac{\Delta a}{a_0} V_0 \tag{27}$$

$$\Delta V_e = \frac{1}{2}\Delta e V_0 \tag{28}$$

$$\Delta V_i = 2V_0 \sin\left(\frac{\Delta i}{2}\right) \tag{29}$$

$$\Delta V_\Omega = \sin i_0 \Delta\Omega V_0 \tag{30}$$

where V_0 is the spacecraft's characteristic velocity with zero eccentricity

$$V_0 = \sqrt{\frac{\mu}{a_0}} \tag{31}$$

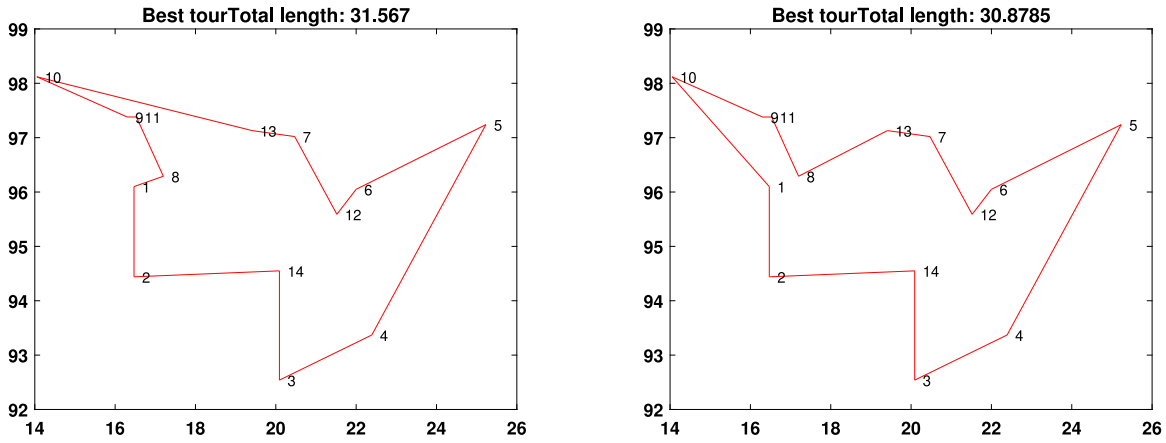

Fig. 5. Benchmark burma14: Test results of proposed TSP algorithm (left) vs. the optimal solution(right).

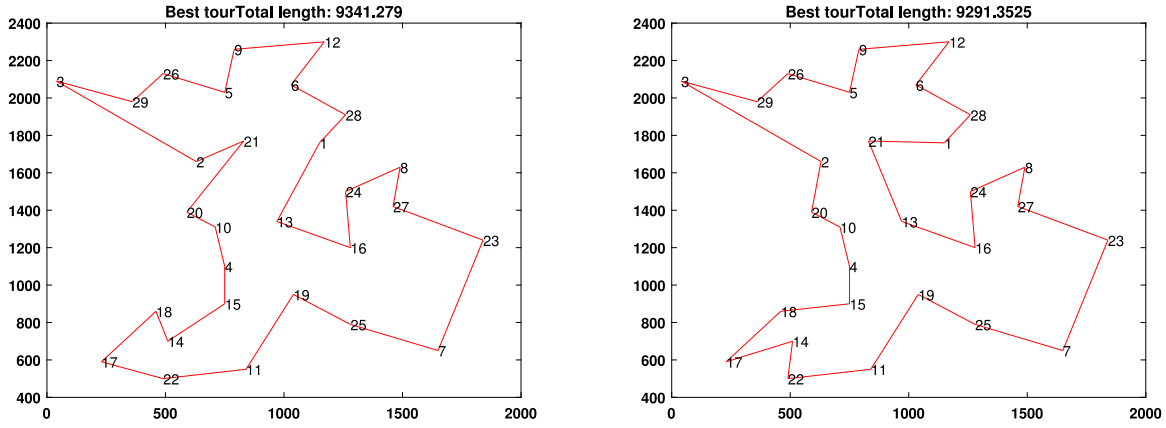

Fig. 6. Benchmark bays29: Test results of proposed TSP algorithm (left) vs. the optimal solution(right).

and

$$\Delta a = \left|a_0 - a_f\right|, \quad \Delta e = \left|e_0 - e_f\right|, \quad \Delta i = \left|i_0 - i_f\right| \tag{32}$$

The desired node drift and drift rate, given transfer time Δt, are given by

$$\Delta\Omega_d = \left|\left(\Omega_f + \dot{\Omega}_f \Delta t\right) - \left(\Omega_0 + \dot{\Omega}_0 \Delta t\right)\right| \tag{33}$$

$$\Delta\dot{\Omega}_d = \frac{\Delta\Omega_d}{\Delta t} \tag{34}$$

where the drift rate of ascending node considering J2 impacts, $\dot{\Omega}_f$ and $\dot{\Omega}_0$ is computed using Eq. (22).

The total cost of the transfer consists of the ΔV components for the semi-major axis Δa, inclination Δi,eccentricity Δe, and ascending node $\Delta\Omega$, and can be computed as

$$\Delta V = \sqrt{\Delta V_a^2 + \Delta V_e^2 + \Delta V_i^2} + \Delta V_\Omega \tag{35}$$

In the above equation, ΔV of phase change is not included, as the transfer time is considered to be long enough for appropriate phasing the satellite. Component of the cost, ΔV_a, ΔV_e and ΔV_i can be computed explicitly using Eqs. (27), (28) and (29), respectively. As for ΔV_Ω, its value varies with both the orbital elements and Time of Flight, several cases are therefore considered in its computation.

4.1.3. To achieve the desired $\Delta\Omega_d$ and $\Delta\dot{\Omega}_d$

Given σ_0, σ_f, along with Δt, one can compute the desired node drift, $\Delta\Omega_d$, using Eq. (33). The node shift $\Delta\Omega_d$ can be achieved by the ΔV either directly using $\Delta\Omega_d$ or indirectly using the $\Delta\dot{\Omega}_d$. In the case of $\Delta\Omega_d$, Eq. (30) can be used to compute the cost. If the node drift rate $\Delta\dot{\Omega}$ is used, some additional cases will be considered since both Δa and Δi can be used for generating $\Delta\dot{\Omega}$.

Given Δa and Δi, the node rate change due to semi-major axis change, $\Delta\dot{\Omega}_a$ and inclination change, $\Delta\dot{\Omega}_i$ can be approximated as [10]

$$\Delta\dot{\Omega}_a = \frac{\partial\dot{\Omega}}{\partial a}\Delta a \approx -\frac{7}{2}\dot{\Omega}_0\frac{\Delta a}{a_0} \tag{36}$$

$$\Delta\dot{\Omega}_i = \frac{\partial\dot{\Omega}}{\partial i}\Delta i \approx -\dot{\Omega}_0 \tan i_0 \Delta i \tag{37}$$

Change of inclination, Δi_Ω, and change of semimajor axis, $\Delta a_{\dot{\Omega}}$ for generating $\Delta\dot{\Omega}$ can therefore be computed as

$$\Delta a_{\dot{\Omega}} = -\frac{2}{7}\frac{\Delta\dot{\Omega}}{\dot{\Omega}_0}a_0 \tag{38}$$

$$\Delta i_{\dot{\Omega}} = -\frac{\Delta\dot{\Omega}}{\dot{\Omega}_0 \tan i_0} \tag{39}$$

Corresponding ΔV can be computed using Eqs. (27) and (29)

$$\Delta V_{a,\dot{\Omega}} = \frac{1}{2}\frac{\Delta a_{\dot{\Omega}}}{a_0}V_0 \tag{40}$$

$$\Delta V_{i,\dot{\Omega}} = 2V_0 \sin\left(\frac{\Delta i_{\dot{\Omega}}}{2}\right) \tag{41}$$

The total ΔV of transferring the spacecraft to the debris consists of two parts. One is used to change the node drift $\Delta\Omega$, and another one is implemented to match the semi-major axis, eccentricity and inclination.

Desired change of the node drift $\Delta\Omega_d$ can be achieved via changing directly the node drift $\Delta\Omega$ (Eq. (30)) or indirectly using the drift rate $\Delta\dot{\Omega}$ (Eqs. (40), (41)).

4.1.4. Cost of the transfer given the elements and time of flight

Given the initial and target elements $\boldsymbol{\sigma}_0$ and $\boldsymbol{\sigma}_f$, the cost of transfer with ToF Δt can be computed. Three cases are considered in the computation:

(1) Compute desired node drift $\Delta\Omega_d$ using the elements $\boldsymbol{\sigma}_0$, $\boldsymbol{\sigma}_f$ and Δt (Eq. (33)), implement ΔV_Ω using Eq. (30) to match the node first, a final $\Delta V_a, \Delta V_i$ and ΔV_e are then implemented to match the semi-major axis Δa, eccentricity Δe and inclination Δi (Eqs. (27)–(29)).

(2) First, Compute $\Delta V_a, \Delta V_e$ and ΔV_i to match the $\Delta a, \Delta e$ and Δi (Eqs. (27)–(29)).$\Delta a, \Delta e$ and Δi are computed using the elements $\boldsymbol{\sigma}_0$ and $\boldsymbol{\sigma}_f$. Considering drift rate of both $\boldsymbol{\sigma}_0$ and $\boldsymbol{\sigma}_f$, given Δt, the total node drift to be matched, $\Delta\Omega_d$ can be obtained using Eq. (33). Compute the drift rate $\Delta\dot{\Omega}_a$ and $\Delta\dot{\Omega}_i$ using $\Delta a, \Delta i$ (Eqs. (36) and (37)), remaining drift of node to be matched after Δt, $\Delta\Omega_r = \Delta\Omega_d - (\Delta\dot{\Omega}_a + \Delta\dot{\Omega}_i)\Delta t$ can be obtained. A ΔV_Ω is implemented to match the remaining node drift $\Delta\Omega_r$ (Eq. (30)).

(3) Given the elements $\boldsymbol{\sigma}_0$, $\boldsymbol{\sigma}_f$ and Δt, compute desired drift rate $\dot{\Omega}_d$ using Eqs. (33) and (34). Compute $\Delta a_{\dot{\Omega}}$ and $\Delta i_{\dot{\Omega}}$ using $\Delta\dot{\Omega}_d$ (Eqs. (38) and (39)), corresponding $\Delta V_{a,\dot{\Omega}}$ (Eq. (40)) and $\Delta V_{i,\dot{\Omega}}$ (Eq. (41)) can be obtained, a final $\Delta V_a, \Delta v_e$ and ΔV_i are then implemented to match the remaining semi-axis, $\Delta a - \Delta a_{\dot{\Omega}}$, inclination $\Delta i - \Delta i_{\dot{\Omega}}$ and eccentricity Δe.

In each case, there are also some sub-cases, e.g. in case 2, the node drift can be changed with Δa or Δi, or both Δa and Δi, while in case 3, both $\Delta a_{\dot{\Omega}}$ and $\Delta i_{\dot{\Omega}}$ can be used. The sub-case with minimum ΔV is used to determine the cost of the case. Velocity increment for each case are computed and compared, among them the minimum ΔV will be used to estimate the body-to-body ΔV.

Note that in some cases of the transfer computation using drift rate, such as case 3, both $\Delta a_{\dot{\Omega}}$ and $\Delta i_{\dot{\Omega}}$ could be used to compute the transfer cost. The plane change is comparatively expensive (about 1.3 km/s per ten degrees in GTOC9 debris removal [15]). The transfer using $\Delta i_{\dot{\Omega}}$ should be usually avoided. However, for larger drift-rate changes, changing inclination is increasingly more effective than raising apoapsis in affecting the node rate [15].

Several cases are considered in estimating ΔV for the rendezvous. Since the approximations are developed using the same dynamics equations, velocity increment can be viewed as a continuous function of ToF and orbital elements, and a derivative of ΔV can be computed as the orbital element varies. In Table 8, the approximate transfer computation results are compared to those obtained using numerical solution. The error of the approximate model is within 10% to those of numerical one.

The approximation can be seen as a scalar cost function given ToF, start and final orbital elements to achieve. A sequence of transfer can then be constructed given a set of targets to fly and corresponding ToF between them. The sequence construction is the traveling sales man problem too, except the dynamics and ToF are considered. After some adaptation, the sequential rendezvous problem could be converted into a continuous parameter estimation.

4.2. Optimization of the dynamic TSP of debris removal

4.2.1. Reformulate the multiple rendezvous problem into a continuous optimization problem

In the dynamic TSP of multiple rendezvous, the optimal sequence $\{\mathcal{I}_i\}$ is determined to minimize the total cost of $\{V_i\}$. With the equation of motion, given $\left[\Delta a_i, \Delta e_i, \Delta i_i, \Delta e_i\right]$ and ToF, T_i with respect to $\mathcal{I}_{i-1}$, the cost of rendezvous to the ith target, V_i, can be determined. The equation below can be used to express the multiple rendezvous:

$$\min_{\mathcal{I}_i} J = \sum_{i=1}^{N} y_i \quad \text{such that } y_i = f\left(T_i, \mathbf{x}_i\right) \tag{42}$$

where y_i is cost of the ith arc, and $\mathbf{x}_i$ is the state $\mathbf{x}_i = \left[\Delta a_i, \Delta e_i, \Delta i_i, \Delta\Omega_i\right]$.

It can be seen that the multiple rendezvous sequence problem is also a discrete sequence optimization problem. Estimate of the target $\mathcal{I}_i$ and the cost y_i is affected by the uncertainty in determining the node $\mathcal{I}_{i-1}$. To associate the integer $\mathcal{I}_i$ with continuous parameters, an expected $\mathcal{I}_i$ can be defined using $\bar{\mathbf{x}}_i$ and $\boldsymbol{\Sigma}_i$. Similar to the static TSP, in optimization of the dynamic TSP, the conditioned expected value $\bar{\mathbf{x}}_{i|i-1}$ and the covariance $\boldsymbol{\Sigma}_{i|i-1} = \mathrm{diag}[\boldsymbol{\sigma}_{i|i-1}]$, where $\boldsymbol{\sigma}_{i|i-1} = \left[\sigma_a, \sigma_e, \sigma_i, \sigma_o\right]_{i|i-1}$ are set to the design variables to be estimated. By using the likelihood metric with respect to $\bar{\mathbf{x}}_{i|i-1}$ and $\boldsymbol{\Sigma}_{i|i-1}$, the ith target $\mathcal{I}_i$ can be selected given the candidate set $\{\mathcal{I}_j\}$. The cost of the ith segment, y_i and associated likelihood, $\mathcal{L}_i$ are computed. The optimal sequence is then determined by minimizing the total transfer cost with accumulated posterior. A gradient-based optimizer is used to search recursively the optimal sequence.

4.2.2. Covariance propagation

In static TSP, relation of the estimated $\mathcal{I}_i$, $\bar{\mathbf{x}}_i$ and $\Sigma_{i|i}$ to its preceding node $\mathcal{I}_{i-1}$ is given as

$$\bar{\mathbf{x}}_i = \bar{\mathbf{x}}_{i|-1}, \quad \boldsymbol{\Sigma}_{i|i} = \boldsymbol{\Sigma}_{i|i-1} + \boldsymbol{\Sigma}_{i-1}$$

where $\boldsymbol{\Sigma}_{i-1}$ is uncertainty due to select of $\mathcal{I}_{i-1}$ and propagated to the ith node. The value equals to the posterior of $\mathcal{I}_{i-1}$ in case of the static TSP. In the dynamic TSP, in contrast, the dynamics and propagation of the states between the targets are considered. With the dynamics, update of the covariance $\boldsymbol{\Sigma}_{i|i}$ is computed as

$$\bar{\mathbf{x}}_i = \bar{\mathbf{x}}_{i|-1}, \quad \boldsymbol{\Sigma}_{i|i} = \boldsymbol{\Sigma}_{i|i-1} + \boldsymbol{\Sigma}^+_{i-1} \tag{43}$$

where $\boldsymbol{\Sigma}^+_{i-1}$ is the uncertainty of $\mathcal{I}_{i-1}$ propagated to the ith node considering the ToF, T_i, and

$$\boldsymbol{\Sigma}^+_{i-1} = \mathbf{H}_i \boldsymbol{\Sigma}_{i-1} \mathbf{H}_i^T \tag{44}$$

where $\boldsymbol{\Sigma}_{i-1}$ is the uncertainty due to the selection of $\mathcal{I}_{i-1}$, and $\mathbf{H}_i$ is the transition matrix of the ith segment given T_i, and the state $\mathbf{x}_{i-1}$

$$\mathbf{H}_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{\partial\dot{\Omega}_i T_i}{\partial a_i} & \frac{\partial\dot{\Omega}_i T_i}{\partial e_i} & \frac{\partial\dot{\Omega}_i T_i}{\partial i_i} & 1 \end{bmatrix} \tag{45}$$

Elements of the transition matrix can be computed either analytically or approximately using the finite difference method.

4.2.3. Optimization of the multiple debris rendezvous with maximum likelihood

The maximum logarithm likelihood metric is used to select the target $\mathcal{I}_i$ from the candidate $\{\mathcal{I}_j\}$

$$\begin{aligned} \min_j \mathcal{L}_j = \min_j & \frac{1}{2}\left(\mathbf{x}_j - \bar{\mathbf{x}}_i\right)\left(\boldsymbol{\Sigma}_{i|i} + \mathbf{V}\right)^{-1}\left(\mathbf{x}_j - \bar{\mathbf{x}}_i\right)' \\ & + \frac{1}{2}\log\left(\det\boldsymbol{\Sigma}_{i|i} + \mathbf{V}\right) \end{aligned} \tag{46}$$

where the matrix $\mathbf{V}_i = \mathrm{diag}\left(\mathbf{v}_i\right)$ is the noise of the system equation. $\bar{\mathbf{x}}_i$ and $\boldsymbol{\Sigma}_{i|i}$ are the expected value and estimated covariance at $\mathcal{I}_i$

$$\begin{aligned} \bar{\mathbf{x}}_i &= \bar{\mathbf{x}}_{i|i-1} \\ \boldsymbol{\Sigma}_{i|i} &= \boldsymbol{\Sigma}_{i|i-1} + \boldsymbol{\Sigma}^+_{i-1} \end{aligned} \tag{47}$$

where $\boldsymbol{\Sigma}^+_{i-1}$ is the covariance due to the uncertain impact of selection at the i-1-th step and propagated to the ith step (Eq. (44)). Posterior after the determination of $\mathcal{I}_i$ is computed as

$$\boldsymbol{\Sigma}_i = \boldsymbol{\Sigma}_{i|i} - \boldsymbol{\Sigma}_{i|i}\mathbf{F}_i^T\left(\mathbf{F}_i\boldsymbol{\Sigma}_{i|i}\mathbf{F}_i^T + \mathbf{W}\right)^{-1}\mathbf{F}_i\boldsymbol{\Sigma}_{i|i}^T \tag{48}$$

where $\mathbf{W} \in \mathbb{R}$ is covariance of the noise of the cost equation, $\mathbf{F}_i$ is the cost Jacobian of the transfer.

In determining the cost to $\mathcal{I}_i$, several subcases are considered. Because the analytical equations are from the same dynamic equations, the cost function can be seen as a continuous function. The Jacobian of the cost, given the vector $\mathbf{x}_i$, can be obtained as

$$\mathbf{F}_i = \left[\frac{\partial V_i}{\partial a_i} \quad \frac{\partial V_i}{\partial e_i} \quad \frac{\partial V_i}{\partial i_i} \quad \frac{\partial V_i}{\partial \Omega_i}\right] \tag{49}$$

With estimated target $\mathcal{I}_i$, logarithm likelihood of the transferring cost to $\mathcal{I}_i$ can be computed, and has the form

$$\begin{aligned}\mathcal{L}_i = &\frac{1}{2}\left(y_i - \bar{y}_i\right)\left(\mathbf{F}_i \boldsymbol{\Sigma}_{i|i}\mathbf{F}_i^T + \mathbf{W}\right)^{-1}\left(y_i - \bar{y}_i\right)^T \\ &+ \frac{1}{2}\log\det\left(\mathbf{F}_i \boldsymbol{\Sigma}_{i|i}\mathbf{F}_i^T + \mathbf{W}\right)\end{aligned} \tag{50}$$

where $y_i = f(T_i, \mathbf{x}(\mathcal{I}_i))$ is the cost to the target $\mathcal{I}_i$, $\bar{y}_i = f(T_i, \bar{\mathbf{x}}(\mathcal{I}_i))$ is expected cost of $\mathcal{I}_i$ using the expected $\bar{\mathbf{x}}_i$.

The prior of the TSP sequence consists of $\{\theta_i\}$ and the ToF series $\{T_i\}$, where θ_i is the conditioned expected values $\left[\bar{\mathbf{x}}_{i|i-1}, \boldsymbol{\sigma}_{i|i-1}\right]$. The objective of the TSP is to minimize the total cost of the TSP and maximize at the same time its accumulated likelihood. The following equation can be used to optimize the whole path

$$\min_{\theta_i, T_i} J = \sum_{i=1}^{N}\left(y_i + \mathcal{L}_i\right) \tag{51}$$

A gradient-based optimizer can be used to search the optimal sequence.

Fig. 7 shows flow chart of proposed dynamic TSP for the multiple rendezvous. Compared to the static TSP, an extra step of covariance propagation based on dynamic equations is required in the dynamic TSP.

4.3. Dynamic TSP with multiple vehicles

Extend the dynamic TSP to the scenario of dynamic TSP with multiple spacecraft. Suppose the number of spacecraft is N_r. The objective of TSP optimization is to determine the sequence $\{\mathcal{I}_i^k\}$ with $i = 1, \ldots, N$ and $k = 1, \ldots, N_r$, and minimize the total tour cost of the vehicles $\sum_{i=1}^{N}\sum_{k=1}^{N_r} y_i^k$, where y_i^k is the cost of transferring the spacecraft k to its ith target $\mathcal{I}_i^k$.

Similar to the TSP with a single spacecraft, the targets can be selected and assigned to the vehicle with a probabilistic ellipsoid. The likelihood of the costs of each vehicle can then be computed and used to select the target on the tours. The total cost of the transfers with accumulated likelihood of the cost are optimized using a gradient-based optimizer.

Suppose at the ith step of the TSP, one has the statistics of each individual vehicle's path as $\left\{\bar{\mathbf{x}}_i^k, \Sigma_i^k\right\}$. Similar to the single-vehicle case, the expected value and covariance of the position conditioned on the preceding step can be computed as

$$\begin{aligned}\bar{\mathbf{x}}_i^k &= \bar{\mathbf{x}}_{i|i-1}^k \\ \boldsymbol{\Sigma}_i^k &= \boldsymbol{\Sigma}_{i|i}^k - \boldsymbol{\Sigma}_{i|i}^k \mathbf{F}_i^{kT}\left(\mathbf{F}_i^k \boldsymbol{\Sigma}_{i|i}^k \mathbf{F}_i^{kT} + \mathbf{W}\right)^{-1}\mathbf{F}_i^k \boldsymbol{\Sigma}_{i|i}^T\end{aligned} \tag{52}$$

where

$$\boldsymbol{\Sigma}_{i|i}^k = \boldsymbol{\Sigma}_{i|i-1}^k + \boldsymbol{\Sigma}_{i-1}^{k,+}$$

is the covariance of spacecraft k due to the uncertain impact of its selection of $\mathcal{I}_{i-1}^k$ and propagated to the ith step. The Jacobian-based transition matrix is used to compute and propagate the covariance $\boldsymbol{\Sigma}_{i-1}^{k,+}$ given T_i and the uncertainty at the $i-1$-th step, Σ_{i-1}^k.

For each individual vehicle, given the set of candidate targets $\{\mathcal{I}_j\}$, similar to the single vehicle TSP, the maximum logarithm likelihood metric concerning expected $\bar{\mathbf{x}}_i^k$ and $\boldsymbol{\Sigma}_{i|i}^k$ is used to select the candidate $\mathcal{I}_i^k$

$$\begin{aligned}\min_j \mathcal{L}_j = \min_j &\frac{1}{2}\left(\mathbf{x}_j - \bar{\mathbf{x}}_i^k\right)\left(\boldsymbol{\Sigma}_{i|i}^k + \mathbf{V}_i\right)^{-1}\left(\mathbf{x}_j - \bar{\mathbf{x}}_i^k\right)' \\ &+ \frac{1}{2}\log\left(\det \boldsymbol{\Sigma}_{i|i}^k + \mathbf{V}_i\right)\end{aligned} \tag{53}$$

Table 6
Parameter settings of the design parameters.

Parameters	Lower bound	Upper bound	Initial value
ToF (days)	0.5	25	20
μ_a (km)	−150	150	0
μ_e	−1.0e−3	1.0e−3	0
μ_i (deg)	−1.5	1.5	0
μ_Ω (deg)	−8.0	8.0	0
σ_a (km)	5.0	50	30
σ_e	1.0e−4	1.0e−3	5.0e−4
σ_i (deg)	0.1	1.0	0.5
σ_Ω (deg)	0.1	8	5

where the matrix $\mathbf{V}_i = \mathrm{diag}\left(\mathbf{v}_i\right)$ is covariance of the noise of the system equation.

As for the cost equation to the estimated $\mathcal{I}_i^k$, associated logarithm likelihood can be computed as

$$\begin{aligned}\mathcal{L}_i^k = &\frac{1}{2}\left(y_i^k - \bar{y}_i^k\right)\left(\mathbf{F}_i^k \boldsymbol{\Sigma}_{i|i}^k \mathbf{F}_i^{kT} + \mathbf{W}\right)^{-1}\left(y_i^k - \bar{y}_i^k\right)' \\ &+ \frac{1}{2}\log\det\left(F_i^k \boldsymbol{\Sigma}_{i|i}^k F_i^{k\prime} + \mathbf{W}\right)\end{aligned} \tag{54}$$

where $\mathbf{W} \in \mathbb{R}$ is the noise covariance of the cost function.

The objective of the TSP is to optimize the total cost of the TSP as well as its corresponding likelihood. The sequence is constructed with the prior, $\theta_i^k = \left[\bar{\mathbf{x}}_{i|i-1}^k, \boldsymbol{\sigma}_{i|i-1}^k\right]$. The following equation can then be used to optimize the whole path with maximum likelihood

$$\min_{\theta_i^k, T_i^k} J = \sum_{i=1}^{N}\sum_{k=1}^{N_r}\left[y_i^k + \mathcal{L}_i^k\left(\theta_i\right)\right] \tag{55}$$

5. Numerical simulation

Consider a removal sequence problem of Sun-Synchronous Low-Earth-Orbit debris. The data set of the debris parameters is from the 9th Global Trajectory Optimization Competition (GTOC9) [27]. Inclinations and semi-major axes of the debris objects are within about 96°–101° and about 600–900 km larger than the Earth's radius, respectively. The eccentricity of the debris orbit ranges from about 0.02 down to almost zero. The task is to design a series of transfers to remove a set of orbiting debris.

5.1. Dynamic TSP with single spacecraft

We first demonstrate the dynamic TSP optimization by implementing the multiple rendezvous with a single spacecraft. The number of candidate debris is 122. In this test, the number of rendezvous targets is set to 13. Table 6 shows the lower and upper bounds of design variables for the TSP optimization. Note that not only the expected orbital parameters of the targets, but their variances are optimized. Both the expected values and variances are distribution parameters conditioned on preceding target.

The TSP starts from the debris $\mathcal{I}_0 = 23$, with the epoch $MJD_0 = 23557.18$. The Matlab SQP solver, fmincon is used to search the optimal sequence. Table 7 shows the results of the multiple rendezvous.

In the optimal TSP path, most of the ToF is close to the upper bound. It looks like with the optimal tour, the transfer takes the longest ToF to take the most advantage of the ascending node's drift to reduce the fuel. The total cost ΔV of the path is 2.610 km/s. With the data set of GTOC9 debris, drift rate of the ascending nodes ranges from 0.75°/day to 1.3°/day [15]; drift-rate changes of about 0.1°/day can be achieved for about 100 m/s ΔV [15]. If the targets on the path are properly paired, difference of the drift rate between the debris can be used for matching the ascending node. The rate $\Delta\dot{\Omega}$ between the pairs could be up to about 0.55°/day, and 13.75° $\Delta\Omega$ for the maximum ToF of 25 days. It would therefore be better to exploit the drift rate over the full period.

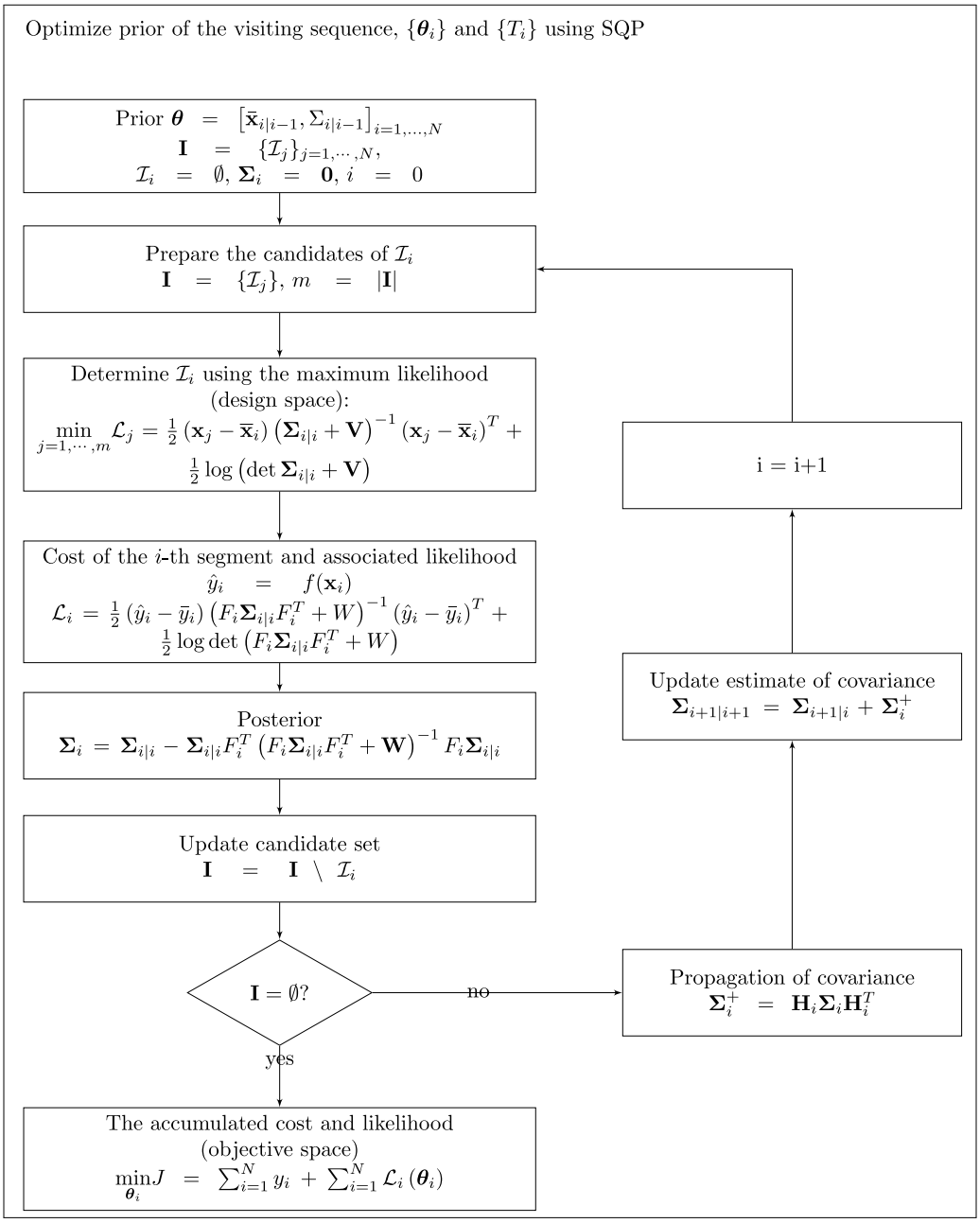

Fig. 7. Flow chart of dynamic TSP of space debris removal.

Table 7
Multiple rendezvous with single spacecraft.

ToF (days)	ΔV (km/s)	$\mathcal{I}_{i-1}$	$\mathcal{I}_i$
25.0	0.25027	23	113
24.999	0.055613	113	79
25	0.072453	79	55
24.999	0.12857	55	121
25	0.21719	121	117
25	0.68523	117	96
25	0.35377	96	118
25	0.056020	118	95
25	0.072048	95	50
25	0.22274	50	27
25	0.038042	27	84
25	0.15176	84	83
25	0.30638	83	20

Table 8 shows the JPL's optimal sub-mission initiated at debris 23 with the same epoch $MJD0$. The number of targets is also 14. The total cost of the path equals 3.066 km/s with the numerical model. In Table 8, the transfer costs using the approximate model are also listed. The total cost is 2.907 km/s, implying an error of 10% compared to the numerical one. Both these costs are more expensive than that obtained using the proposed method.

The GTOC-9 is a 10-spacecraft routing problem in succession over an 8-year period. The length of each path is variable and should be optimized. It is mathematically a vehicle capacity routine problem, having more combinatorial constraints than the TSP studied in the test. The method proposed in this work is designed for the fixed length TSP with specified numbers of targets. No vehicle capacity constraint is considered. After some adaptation and mathematics formulations, the original GTOC-9 problem could be potentially reformulated as continuous parameter estimation and optimized with gradient-based searches. Such mathematical formulations will be considered in future research.

Table 8
Multiple rendezvous with single spacecraft (JPL solution [15]).

Parameter	Values
Debris ID	23, 55, 79, 113, 25, 20, 27, 117, 121, 50, 95, 102, 38, 97
ToF, days	24.86, 24.98, 22.42, 24.99, 0.29, 10.63, 25.00, 2.70, 1.51, 1.41, 24.67, 24.31, 5.86
Rendezvous duration, days	5.00, 5.00, 5.04, 5.01, 5.01, 5.03, 5.00, 5.00, 5.00, 5.03, 5.03, 5.04, 5.04, 5.00
ΔV, km/s	0.1618, 0.1392, 0.0658, 0.2082, 0.1152, 0.3001, 0.5649, 0.0783, 0.1050, 0.2333, 0.4535, 0.3404, 0.3008
ΔV, km/s (Analytical model)	0.1598, 0.1299, 0.0581, 0.1746, 0.1181, 0.1966, 0.5644, 0.0622, 0.1075, 0.2447, 0.4468, 0.3414, 0.3019

Table 9
Multiple rendezvous with single spacecraft (JPL solution [15]).

ID	ID	ToF (days)	Dv (km/s)	Dot-O (1/2/3)	ΔV_{Ω} (1/2)	$\Delta\dot{\Omega}_a$ (1/2)	Dvi (1/2)	Direct/Indirect (1/2)
23	55	24.86	0.159891	1	1	2	2	1
55	79	24.98	0.129932	1	2	2	2	2
79	113	22.42	0.056639	2	2	2	2	1
113	25	24.99	0.174656	3	2	2	2	2
25	20	0.29	0.118168	1	1	1	1	2
20	27	10.63	0.196646	3	1	1	1	1
27	117	25	0.564488	3	1	1	1	1
117	121	2.7	0.062237	1	1	2	2	1
121	50	1.51	0.107519	1	1	1	1	1
50	95	1.41	0.244718	3	1	1	1	2
95	102	24.67	0.446879	2	1	2	2	1
102	38	24.31	0.341455	2	1	2	2	1
38	97	5.86	0.301962	3	1	1	1	1

The discrete type space TSP with nondifferentiable cost function is under consideration as well (see Table 9).

Debris	Debris	ToF (days)	Analytical (km/s)	Dot-O (1/2/3)	Dvo (1/2)	Direct/ Indirect (1/2)
23	55	24.86	0.1598	1	1	1
55	79	24.98	0.1299	1	2	2
79	113	22.42	0.0566	2	2	1
113	25	24.99	0.1746	3	2	2
25	20	0.29	0.1181	1	1	2
20	27	10.63	0.1966	3	1	1
27	117	25	0.5644	3	1	1
117	121	2.7	0.0622	1	1	1
121	50	1.51	0.1075	1	1	1
50	95	1.41	0.2447	3	1	2
95	102	24.67	0.4468	2	1	1
102	38	24.31	0.3414	2	1	1
38	97	5.86	0.3019	3	1	1

5.2. Dynamic TSP with multiple spacecraft

We then extend the test case of dynamic TSP to the sequential debris removal with multiple spacecraft. Three spacecraft are considered to rendezvous and remove the debris. Parameter settings are same as for the single spacecraft case as Table 6 shows. Each sequence of the TSP is set equal to 10 debris, and the maximum ToF is 30 days. The initial target and epoch of the TSP are set to same as the single spacecraft case, i.e. $\mathcal{I}_0 = 23$ and $MJD_0 = 23557.18$.

Debris visits of three spacecraft are separated by assigning different ascending nodes. Three TSP paths are then determined and constructed. The lower and upper bounds of the ascending node difference between the three paths $\Delta\Omega_0^{(1,2)}$, $\Delta\Omega_0^{(2,3)}$ are set to 0.2 deg and 5.0 deg respectively. The initial values of $\Delta\Omega_0^{(1,2)}$ and $\Delta\Omega_0^{(2,3)}$ are set to 1.5 deg and 3.5 deg, respectively.

The multiple-vehicle dynamic TSP path planning is implemented with the TSP adaptations as shown in Section 4.3. Tables 10–12 show the simulation results of the TSP. The total cost of the TSP is equal to about 8.283 km/s. Similar to the debris removal with a single spacecraft, in these TSPs, the ToF of each transfer is closed to the upper bound of the TOF to take advantage of the ascending node drift.

Table 10
Multiple rendezvous with multiple spacecraft(A).

ToF (days)	ΔV (km/s)	$\mathcal{I}_{i-1}$	$\mathcal{I}_i$
30	0.159891	23	55
30	0.867595	55	16
30	0.081637	16	118
30	0.233033	118	57
30	0.188833	57	95
30	0.192954	95	38
30	0.078206	38	103
30	0.461527	103	62
30	0.290484	62	97
30	0.133784	97	87

6. Conclusion

TSP is one of the most extensively studied yet very challenging problem to be resolved in the field of computational science. The space TSP design is further complicated by the involved dynamics. In this work, we proposed a likelihood analyzing method to reformulate the combinatorial TSP into continuous parameter estimation, simplifying the discrete branch operation into a serial mode searching. No branch and pruning operations are implemented during the sequence optimization. No population based evolution operation is involved as well. Effectiveness and efficiency of the proposed method are tested through a set of static and dynamic TSP.

The method is designed on a theoretical basis that an optimal path should be closed to its expected values not only in the design

Table 11
Multiple rendezvous with multiple spacecraft(B).

ToF (days)	ΔV (km/s)	$\mathcal{I}_{i-1}$	$\mathcal{I}_i$
30	0.225081	23	113
30	0.366646	113	121
30	0.109776	121	96
30	0.117785	96	25
30	0.181661	25	9
30	0.085561	9	3
30	0.266912	3	20
30	0.084088	20	117
30	0.101098	117	27
30	0.601065	27	84
30	0.225081	23	113
30	0.366646	113	121
30	0.109776	121	96

Table 12
Multiple rendezvous with multiple spacecraft(C).

ToF (days)	ΔV (km/s)	$\mathcal{I}_{i-1}$	$\mathcal{I}_i$
30	0.2598547	23	79
30	0.8873094	79	50
30	0.3682732	50	15
30	0.3693598	15	46
30	0.0942381	46	114
30	0.1994587	114	22
30	0.6231391	22	120
30	0.3480821	120	81
30	0.0404974	81	105
30	0.2649154	105	116
30	0.2598547	23	79
30	0.8873094	79	50
30	0.3682732	50	15

space but the objective evaluations, given the path segment parameters conditioned on its preceding decisions. The likelihood-based metrics and their propagation with respect to the expected path are therefore designed and used to model the impacts of the sequential decision making. Such ideas are designed and compared to the mechanism of the serial parameter estimation in computational science. With the method, the path determination is expressed in a serial mode rather than the branch exploring as most conventional methods did. The results of the tests show that the solution closed to the optimal sequence could be obtained.

In this work, a gradient optimizer is used to search the sequence. The method works efficiently in the tested benchmark and debris removal mission design problems. We hope that in the future the efficiency could be further improved, and could be extended to tackle the large scale dynamic TSP. Potential applications of the method include the TSP with a discrete cost function too, given a new form distance metric which maps the discrete cost function into a continuous space. The method with the discrete cost function is currently considered in our research work.

We also note that the TSP-like problems are encountered in many other mission design problems, such as multiple planets and asteroid mission design, mega-constellation station keeping, etc. After some adaptation, the proposed method might be able to be used to resolve these dynamic combinatorial optimization problems. Some other complex combinatorial optimization problems, such as vehicle capacity routing problems and multi-objective TSP are also under consideration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The research is supported by National Natural Science Foundation of China, No. U20B2056, U20B2054.

References

[1] W.J. Cook, In pursuit of the traveling salesman, in: In Pursuit of the Traveling Salesman, Princeton University Press, 2011.
[2] G. Laporte, The traveling salesman problem: An overview of exact and approximate algorithms, European J. Oper. Res. 59 (2) (1992) 231–247.
[3] M. Shehab, A.T. Khader, M.A. Al-Betar, A survey on applications and variants of the cuckoo search algorithm, Appl. Soft Comput. 61 (2017) 1041–1059.
[4] R. Thamilselvan, P. Balasubramanie, A genetic algorithm with a tabu search (GTA) for traveling salesman problem, Int. J. Recent Trends Eng. 1 (1) (2009) 607.
[5] J.M. Longuski, S.N. Williams, Automated design of gravity-assist trajectories to mars and the outer planets, Celestial Mech. Dynam. Astronom. 52 (1991) 207–220.
[6] A.F. Heaton, N.J. Strange, J.M. Longuski, E.P. Bonfiglio, Automated design of the europa orbiter tour, J. Spacecr. Rockets 39 (1) (2002) 17–22.
[7] A.E. Petropoulos, J.M. Longuski, E.P. Bonfiglio, Trajectories to jupiter via gravity assists from venus, earth, and mars, J. Spacecr. Rockets 37 (6) (2000) 776–783.
[8] Y.-F. Liao, D.-H. Yau, C.-L. Chen, Evolutionary algorithm to traveling salesman problems, Comput. Math. Appl. 64 (5) (2012) 788–797.
[9] M. Ceriotti, M. Vasile, MGA trajectory planning with an ACO-inspired algorithm, Acta Astronaut. 67 (9–10) (2010) 1202–1217.
[10] K. Deb, N. Padhye, G. Neema, Interplanetary trajectory optimization with swing-bys using evolutionary multi-objective optimization, in: Advances in Computation and Intelligence: Second International Symposium, ISICA 2007 Wuhan, China, September 21-23, 2007 Proceedings 2, Springer, 2007, pp. 26–35.
[11] A. Gad, O. Abdelkhalik, Hidden genes genetic algorithm for multi-gravity-assist trajectories optimization, J. Spacecr. Rockets 48 (4) (2011) 629–641.
[12] D. Hennes, D. Izzo, Interplanetary trajectory planning with Monte Carlo tree search, in: Twenty-Fourth International Joint Conference on Artificial Intelligence, 2015.
[13] H.-X. Shen, T.-J. Zhang, L. Casalino, D. Pastrone, Optimization of active debris removal missions with multiple targets, J. Spacecr. Rockets 55 (1) (2018) 181–189.
[14] A.-y. Huang, Y.-z. Luo, Global optimization of multiple-spacecraft rendezvous mission via decomposition and dynamics-guide evolution approach, J. Guid. Control Dyn. 45 (1) (2022) 171–178.
[15] P. Anastassios, G. Daniel, J. Drew, L. Gregory, N. Austin, R. Javier, S. Juan, S. Jeffrey, A. Nitin, P. Thomas, et al., GTOC9: Results from the jet propulsion laboratory (team JPL), Acta Futura 11 (2018) 25–35.
[16] N. Zhang, S. Chen, Z. Zhang, H. Baoyin, Two-stage dynamic-assignment optimization method for multispacecraft debris removal, J. Guid. Control Dyn. 45 (9) (2022) 1750–1759.
[17] A. Barea, H. Urrutxua, L. Cadarso, Large-scale object selection and trajectory planning for multi-target space debris removal missions, Acta Astronaut. 170 (2020) 289–301.
[18] J. Bang, J. Ahn, Multitarget rendezvous for active debris removal using multiple spacecraft, J. Spacecr. Rockets 56 (4) (2019) 1237–1247.
[19] N. Zhang, Z. Zhang, H. Baoyin, Timeline club: An optimization algorithm for solving multiple debris removal missions of the time-dependent traveling salesman problem model, Astrodynamics (2022) 1–16.
[20] T. Uriot, D. Izzo, L.F. Simões, R. Abay, N. Einecke, S. Rebhan, J. Martinez-Heras, F. Letizia, J. Siminski, K. Merz, Spacecraft collision avoidance challenge: Design and results of a machine learning competition, Astrodynamics 6 (2) (2022) 121–140.
[21] L. Hou, Z. Hou, H. Ma, Y. Yang, Optimal multi-gravity-assist trajectories design with likelihood analysis, in: 2018 IEEE Congress on Evolutionary Computation (CEC), IEEE, 2018, pp. 1–8.
[22] G. Reinelt, Tsplib95, Vol. 338, Interdisziplinäres Zentrum für Wissenschaftliches Rechnen (IWR), Heidelberg, 1995, pp. 1–16.
[23] A.-Y. Huang, Y.-Z. Luo, H.-N. Li, Fast estimation of perturbed impulsive rendezvous via semi-analytical equality-constrained optimization, J. Guid. Control Dyn. 43 (12) (2020) 2383–2390.
[24] S. Chen, H. Baoyin, Analytical estimation of the velocity increment in j 2-perturbed impulsive transfers, J. Guid. Control Dyn. 45 (2) (2022) 310–319.
[25] M. Cerf, Multiple space debris collecting mission—debris selection and trajectory optimization, J. Optim. Theory Appl. 156 (3) (2013) 761–796.
[26] T.N. Edelbaum, Propulsion requirements for controllable satellites, ARS J. 31 (8) (1961) 1079–1089.
[27] D. Izzo, M. Märtens, The kessler run: on the design of the GTOC9 challenge, Acta Futura 11 (1) (2018) 11–24.