

# 一种多状态空间信息网络拓扑生成算法的优化



杨 鹏, 张嘉颖, 周世杰\*, 周湘阳

(电子科技大学 数学科学学院, 成都 611731)

**摘要** 空间信息网络是一种具有节点运转高速性、周期性的网络。随着近地轨道卫星日益增多, 空间信息网络拓扑动态性极强, 网络拓扑抗毁优化问题将具有研究意义。在考虑卫星组网的可视性、卫星节点的连接度、以及整个网络通信链路数等多种状态情况下, 以最小化网络中卫星节点间的端到端时延为优化目标, 构建一个满足多种约束条件的网络拓扑优化模型, 提出一种优化后的模拟退火算法对模型进行求解, 在模拟退火过程中创新性的提出了网络流算法进行邻域求解。实验表明, 模拟退火混合求邻域算法显著优于模拟退火随机求邻域算法。

**关键词** 空间信息网络; 网络拓扑动态优化; 网络流算法; 模拟退火算法

中图分类号 TP311 文献标志码 A DOI 10.12178/1001-0548.2022377

## An Optimization Method in Multi-State Spatial Information Network Topology Generation

YANG Peng, ZHANG Jiaying, ZHOU Shijie\*, and ZHOU Xiangyang

(School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu 610000, China)

**Abstract** Spatial information network is a kind of network with high-speed and periodically running nodes. With the increasing number of low Earth orbit satellites, the topology of spatial information networks is highly dynamic, and the problem of network topology survivability optimization will be of great research significance. Considering the visibility of satellite networking, the connectivity of satellite nodes, and the number of communication links in the entire network, a network topology optimization model satisfying multiple constraints is constructed to minimize the end-to-end delay among satellite nodes in the network, and then an optimized simulated annealing algorithm is proposed to solve the model. In the simulated annealing process, the network flow algorithm is innovatively proposed to solve the neighborhood. The experimental results show that the simulated annealing hybrid neighborhood algorithm is significantly better than the simulated annealing random neighborhood algorithm.

**Key words** spatial information network; dynamic optimization of network topology; network flow algorithm; simulated annealing algorithm

随着人类科技和生产力的逐步发展, 人类对于外太空的探索 and 开发也在逐渐提上日程。近来, 无论是马斯克的 SpaceX 还是我国的小行星监测预警系统和近地小行星防御系统计划, 都将要发射更多的卫星, 使得空间信息网络的卫星数量和节点运转复杂性快速上升。相比空间信息网络, 对地面网络的抗毁性研究已取得诸多的成果。由于高空轨道中的卫星网络节点具有高速动态移动的特征, 所以网络拓扑鲁棒性充斥着不确定, 使得空间网络通信的复杂程度远高于地面网络。而地面上的任意两点都能通过高空网络中的一些卫星节点进行快速通信,

所以提供更加稳定高效的通信链接具有现实意义。

与地面网络相比, 空间信息网络具有覆盖范围广、组网灵活等优点<sup>[1]</sup>。与此同时, 由于卫星网络固有的脆弱性, 可能存在一些卫星节点或星间链路突发性失效的情况, 一旦出现故障, 将可能导致整个空间信息网络的通信能力急剧下降, 甚至发生多米诺骨牌效应导致全网中断瘫痪<sup>[2-6]</sup>。在此情况下, 由于太空中的辐射等影响, 卫星所携带的处理器功率不高, 计算算力十分有限, 则更需要设计一个高效的算法来解决网络拓扑结构的快速构造问题。

目前针对空间信息网络构建一个高效、稳定的

收稿日期: 2022-11-05; 修回日期: 2023-03-20

基金项目: 四川省科技计划重点研发项目 (2023YFG0155)

作者简介: 杨鹏, 博士, 副教授, 主要从事空间信息网络方面的研究。

\*通信作者 E-mail: shzhou@uestc.edu.cn

网络拓扑结构, 可以使得卫星之间点对点的时延缩短、抗毁性增强、通信链路成本降低。如文献 [7] 在单星星上在线任务调度规划中提到了基于空间信息网络的各种精确算法和近似算法的研究, 文献 [8] 提出了在多种约束条件限制下把多个时间片当做状态考虑的模拟退火算法, 文献 [9] 提出了基于单时间片的贪心算法来解决星间链路分配问题, 而后, 文献 [10-11] 针对多目标多状态拓扑结构生成模型提出了改进的模拟退火优化算法, 随后基于单时间片的调度算法<sup>[12]</sup>、负载均衡算法<sup>[13]</sup>、免疫算法<sup>[14]</sup>等都相继被提出。文献 [15] 提出了基于空间信息网络抗毁模型的多时间片优化算法, 文献 [16] 则在文献 [11] 的基础上改进了邻域解求取方式, 进一步优化了模拟退火算法, 文献 [17] 针对空间信息网络多目标抗毁模型提出了改进的 NSGA 遗传算法。

本文将基于单时间片的模拟退火算法进行改进, 以最小化动态网络拓扑结构的平均点对点时延为目标, 同时考虑尽可能缩短星上运算时间和计算能力的消耗, 以保证空间信息网络的抗毁性。

## 1 空间信息网络模型

在空间信息网络模型中, 所有的卫星节点会沿着各自的轨道一直保持高速运转, 致使节点之间的距离和相对位置会不断的发生变化。所以可以把卫星网络的整个运转周期分成若干个时间片, 并在每个时间片内分别进行计算, 而且每一个时间片都对

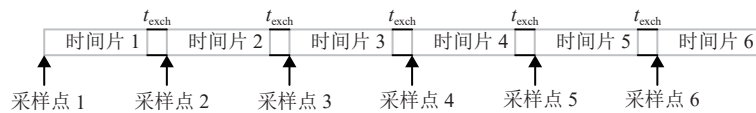


图 1 等长时间间隔时间片划分

空间信息网络的时延由 4 个部分组成: 上行链路时延、下行链路时延、卫星间传输时延、单星内部计算与处理时延。本文主要研究星间网络, 不考虑地面对卫星的影响, 因此忽略上行链路时延和下行链路时延。

由于卫星是按照固定的轨道运行, 因此卫星之间是否可视可以经由一系列计算得到。而当两颗卫星互相可视时, 即可简单的按照星间直线距离除以光速作为两颗卫星之间的单次信息传输的星间传输时延。在保证由卫星组成的网络总体连通的情况下, 则网络中的任意两个节点都可以作为一次信息传输的起点或终点。信息传输过程中的总时延可以

应着一段时间的网络空间拓扑结构。在传统的时间片分割方法中, 一个时间片的长度通常会选择 15~120 s 的某个时间值。这样的一个时间段内的网络拓扑结构会被称为一个网络拓扑快照。在一个时间片内的网络拓扑快照是唯一的, 这也就代表一个时间片中网络拓扑结构是被认为固定不变的。通过这种方式可以把时间划分成若干个。而相邻两个时间片之间也会存在中间状态, 在中间状态中, 上一个时间片中的一些卫星节点间的链路断开 (即物理上不再可见), 而一些新的链路又会被建立。当卫星节点间的通讯链接的变化完成后, 新的拓扑快照就会形成, 此时就将进入下一个时间片。

这种时间片划分方式需要实时检测空间信息网络是否发生了节点链接的更换, 一旦发生了旧链接的断开和新链接的建立, 则需要立即把一个时间片划分开, 这样会导致算法运行的效率降低。由于空间信息网络拓扑结构是高速动态变化的, 这种划分方式会导致其划分出大量的时间碎片, 也同样影响了其他环节的计算性能。因此, 本文将采用等长时间间隔划分法进行时间片的划分。即把时间划分成等长的若干个时间片, 然后在每个时间片的初始时刻进行数据采样, 用采样得到的网络拓扑结构充当这一个时间片的网络拓扑快照。由于空间信息网络的运行特性, 相邻的两个时间片的拓扑结构差别非常细微, 因此可以近似表示当前这个时间片的拓扑快照。用这种方法得到的拓扑快照将兼具快速与简洁的优点。等长时间间隔片划分如图 1 所示。

近似认为是起点到终点这条路径上若干次传输时延和路径上每一颗卫星的单星内部计算与处理时延的总和。因此可以使用最短路模型计算任意两节点间的传输时延。

若某一条路径经过的节点按顺序设为  $p_1, p_2, p_3, \dots, p_n$ , 设单星计算与处理时延为  $T_{Process}$ , 由于同一个星座中的卫星通常算力是比较接近的, 则这条路径在某个时间片  $A$  中的总时延可以表示为:

$$\text{delaysum}(A, p_1, p_k) = \begin{cases} \text{Length} \times T_{process} & \text{无边权} \\ \sum_{i=1}^{k-1} \text{delay}(A, p_i, p_{i+1}) & \text{有边权} \end{cases} \quad (1)$$

式中, Length 是路径的长度。

对于早期的卫星, 其单星内部计算与处理时延较大, 因此在计算时应优先考虑自身的计算处理能力, 所以将采用无边权模型进行计算, 尽量减小卫星间传输时延对结果的影响, 使得路径经过更少的卫星。此时总时延便与星间距离无关。针对无边权模型, 由于给定的网络时延参数  $T_{Process}$  是常数, 所以不妨把互相可视的卫星节点之间的边权设为 1, 而不可视的卫星节点之间的边权设为无穷大。

而对于较为先进的卫星, 其算力通常更高, 使得每个卫星节点的计算与处理时延大幅减小, 此时应更侧重于网络中的卫星节点间的传输时延, 所以将采用有边权模型进行计算, 此时将采用两节点间的物理距离除以光速作为边权值。对于不可视的两颗卫星节点, 则以一个近似无穷大的数作为边权用以消除影响。

## 2 卫星网络模型

设矩阵  $V$  为可视矩阵,  $v_{i,j}$  代表在该时间片中  $i, j$  两个节点之间的距离。在无边权模型中, 两节点可见, 则有  $v_{i,j} = 1$ , 否则,  $v_{i,j} = \infty$ 。在有边权模型中,  $v_{i,j}$  代表两个卫星节点的直接通信时延。设矩阵  $C$  为链路选择矩阵, 若两节点已建立通信链接, 则  $c_{i,j} = 1$ , 否则为 0。在此基础上, 求出一个子网络拓扑使得节点之间的平均传输时延最小便成了一个优化问题。设函数  $d_{i,j}(V, C)$  为节点  $i$  至节点  $j$  的最短路径长度。具体数学优化模型为:

$$\min \tau(A) = \frac{2}{n(n+1)} \sum_{i=1}^n \sum_{j=1}^i d_{i,j}(V, C)$$

$$\text{s.t.} \begin{cases} \sum_{i=1}^n c_{j,i} \leq \max \text{ degree} \quad \forall j \in \{1, 2, \dots, n\} \\ \sum_{i=1}^n \sum_{j=i+1}^n c_{i,j} \leq \max \text{ edge} \\ c_{i,j} = 0, \quad \forall v_{i,j} = \infty \\ c_{i,j} \in \{0, 1\} \quad \forall v_{i,j} = 1 \end{cases} \quad (2)$$

式中,  $\tau$  为整个网络中所有卫星节点通信传输时延的平均值;  $\max \text{ degree}$  代表着网络中的每一个卫星节点所能建立的链路上限;  $\max \text{ edge}$  代表整个网络中通信链路的总和的上限。

## 3 模拟退火算法求解模型

模拟退火算法是一种经典的启发式经典算法。模拟退火算法并不限制解的形式, 只需提出目标函

数和约束条件, 以及构造一个求邻域解的方法, 即可有效地在较低的时间复杂度下找到全局最优解。因此, 模拟退火算法在诸多优化问题上都提供了较为高效的求解。

模拟退火的核心思路是模仿金属冷却的过程。在锻造金属的过程中, 金属的温度会经历由高变低的冷却过程, 而金属的内能也会由高变低。对应到算法中, 即为设定一个对应的温度  $T$ , 通过邻域搜索找到一个邻域解之后把新的邻域解与当前解进行对比。设当前解为  $E_t$ , 新的邻域解为  $E_{t+1}$ , 则接受新的邻域解的概率为:

$$P = \begin{cases} 1 & E_{t+1} < E_t \\ e^{-\frac{E_t - E_{t+1}}{kT}} & E_{t+1} > E_t \end{cases} \quad (3)$$

式中,  $k$  为任意设定常数。由式 (3) 可见, 存在一定的概率使得模拟退火算法会接受相对更差的解。而这个概率会随着温度的降低而逐渐降低。并且如果新的邻域解与当前解相比相差较多, 则接受这个相对更差的解的概率也会变小。而随着时间的推进温度会逐渐减小, 直至归零。因此, 这种能够接受更差解的逻辑使得模拟退火算法得以保证寻找到全局最优解。算法流程如图 2 所示。

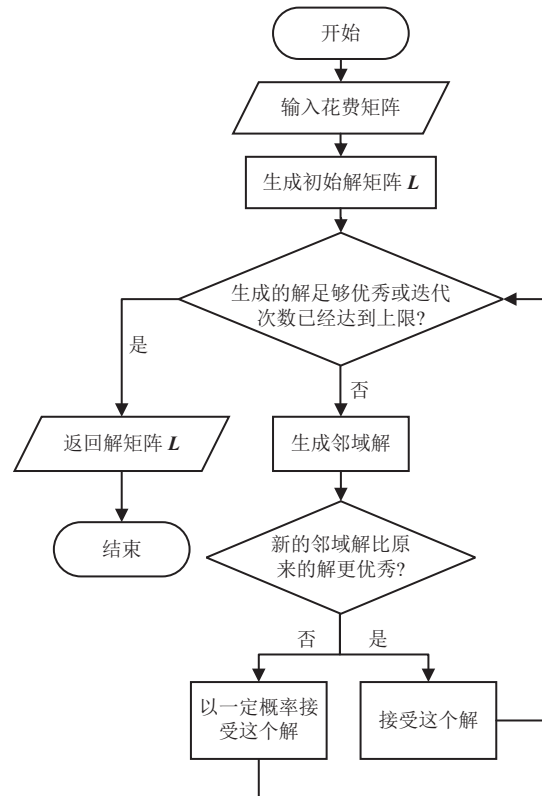


图 2 模拟退火算法流程图

### 3.1 初始解的选取

在模拟退火的过程中, 首先需要生成一个初始解矩阵  $L$ 。初始解矩阵  $L$  的生成必须满足卫星网络模型中约束条件的限制, 具体选取流程如图 3 所示。

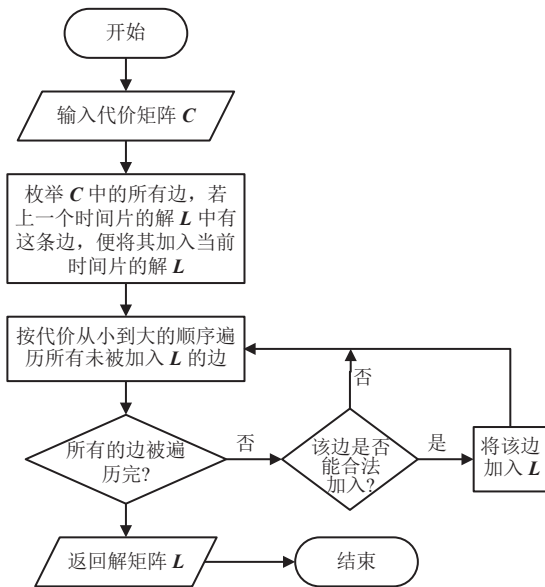


图 3 初始解选取流程图

### 3.2 邻域解的求取

邻域解的选取, 需要在保证空间信息网络完全连通的情形下进行。由于在网络不连通的时候, 处于两个不同连通块内的卫星节点之间的距离会被设置成无穷远。此时求解出来的节点间的平均时延也会变成无穷大, 这就保证了这个解不会被模拟退火算法选择为新解。所以在邻域求解的过程不需要考虑如何维护空间信息网络的连通性, 而只需要限制卫星节点链接数量即可。

一种方式是采用生成树随机遍历方式<sup>[16]</sup>求解邻域解, 其核心思想是把一条链连成环, 然后随机删除环中的某一条边。首先算法会随机选择两个节点  $s, t$ , 并保证这两个节点的度数都未达到上界, 然后以  $s$  为起点对整个网络进行深度优先搜索, 将会得到一棵深度优先遍历树。另一种方式是采用双链路随机交换方式<sup>[11]</sup>, 该方法求取过程更为简洁。

对于生成树随机遍历方式求邻域解, 在确定了两个节点  $s, t$  的情况下, 找到节点  $s$  到  $t$  的一条简单路径的时间复杂度仅为  $O(n)$ , 而从中随机选择一条边删除时间复杂度是  $O(1)$ 。对于双链路随机交换方式求邻域解, 同样可以发现在确定了四元组的情况下, 只需要  $O(1)$  的操作即可完成加边与删边操作。

每一次迭代求解的过程中, 除了寻找新的邻域解, 还需要对目标函数部分进行求解。在本文中, 解的目标函数被定义为求拓扑网络的节点之间的最小平均时延。计算过程中需要求取所有节点间的最短路, 计算过程是  $O(n^3)$ 。由于每次求邻域解的主要时间复杂度的瓶颈就在求目标函数上, 因此每次求邻域解的复杂度是  $O(n^3)$ 。

设  $k$  为总迭代步数, 则模拟退火算法的迭代步数取决于初始温度  $T$ 、温度的衰减率  $\Delta T$ 、及退出温度  $\varepsilon$  等几项关键数值, 具体为:

$$\begin{cases} T \times \Delta T^k = \varepsilon \\ k = \log_{\Delta T}(\varepsilon) - \log_{\Delta T}(T) \end{cases} \quad (4)$$

因此模拟退火算法的具体运行时间取决于迭代步数。在本问题中, 可以用解的收敛速度来缩短模拟退火迭代的步数。即选取可视矩阵  $V$  直接求解平均点对间最短时延, 然后以其作为理论最优解  $\tau_{\min}$ , 然后选定一个数值作为退出误差  $\delta$ 。当解矩阵  $L$  满足式 (5) 时, 则直接退出模拟退火。如此即可减少算法迭代的步数。

$$\tau_L = \delta \tau_{\min} \quad (5)$$

#### 3.2.1 网络流邻域求解法

设定一个有向图  $G = (V, E)$ , 图中每一条边  $\langle u, v \rangle$  都存在一个容量  $c(u, v) > 0$ , 则称这张图是一个流网络图。设定图中的两个特殊的节点  $s \in V$  和  $t \in V$  分别为有向图  $G$  的源点和汇点, 且满足  $s \neq t$ 。

设定  $f(u, v)$  为二元组  $(u \in V, v \in V)$  的实数函数, 并称这个函数为流量函数, 则式 (6) 即为一个最大流的优化模型。

$$\begin{aligned} \max F &= \sum_{(s,v) \in E} f(s,v) = \sum_{(u,t) \in E} f(u,t) \\ \text{s.t.} &\begin{cases} 0 \leq f(u,v) \leq c(u,v) & c(u,v) > 0 \\ c(u,v) \leq f(u,v) \leq 0 & c(u,v) < 0 \\ f(u,v) = -f(v,u) \\ \sum_{(u,x) \in E} f(u,x) = \sum_{(x,v) \in E} f(x,v) \forall x \in V - \{s,t\} \end{cases} \end{aligned} \quad (6)$$

经典的网络流 Dinic 算法<sup>[18]</sup>是求取流网络图最大流的一个经典算法, Dinic 算法在图中的每一个顶点都设置了一个当前弧用以存储当前已经被遍历过的边, 并且在每一次最短路径上运行完成后使用多路增广, 用以保证其达到时间复杂度上界。本文采用 Dinic 作为网络流算法, 其运行流程如图 4 所示。

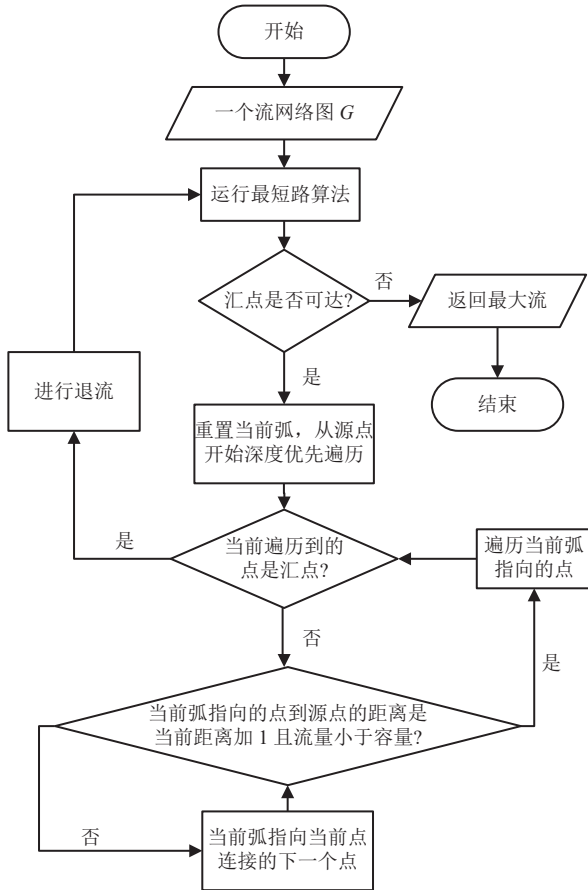


图 4 Dinic 算法流程框图

对于一个连通图  $G = (V, E)$ ，其边连通度为其最小割的大小。通常，一个图的连通度越大，则代表该网络越稳定，其抗毁能力越强。因为最小割的大小可以代表从一个节点到另一个节点的不相干路径数量。不相干路径数量越多，则可认为其遭受攻击之后的继续保持通信及被修复的能力就越强。

在本文中，虽然需要优化的目标函数是节点间平均最小时延，而不是抗毁性。但是抗毁性的增加同样在一定程度上有利于节点间平均最小时延的缩短。一个网络的抗毁性越强，则两节点间的路径就越多，则其几何拓扑表现会趋向于一个超球体，而一个超球体可以最小化点对间的时延。所以本文提出了一种基于网络流的邻域求解法，使得模拟退火在每一次寻找邻域解的时候都可以保证其边连通度不减。

本节的邻域求解算法，首先会随机选择网络中度数小于上限且互相可见的两个卫星节点。然后以这两个节点分别为源点和汇点完成一次最大流最小割的运算。此时网络中没有满流的边即为不会影响网络最小割的边。然后随机寻找一条边删除，并连接源点和汇点，就可以作为新的邻域解。具体的运

行流程如图 5 所示。

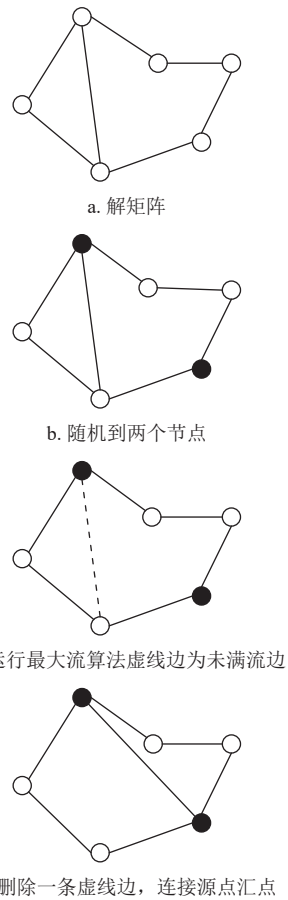


图 5 邻域求解算法的运行流程

在一次邻域求解的过程中，随机找到两个度数小于上限且相互可见的节点所需的时间复杂度是  $O(n^2)$ ，完成最大流运算的时间复杂度为  $O(n^2m)$ 。最后求取节点间平均最小时延的时间复杂度为  $O(n^3)$ 。因此本文提出的基于网络流邻域求解法与生成树随机遍历方式、双链路随机交换方式复杂度同级，均为  $O(n^3)$ 。

### 4 自适应方法

模拟退火算法对于空间信息网络相邻时间片之间的网络拓扑结构的连续性动态改变没有好的应对策略。在空间信息网络中，相邻时间片之间的时间间隔都是以秒为单位。譬如在本文的实验部分，相邻时间片的间隔就选择为 60 s。虽然卫星在不断的高速移动，但是相邻时间片中网络的可视矩阵并没有发生明显的变化。在此情况下，下一个时间片的初始解就可以利用上一个时间片的解的信息生成一个新的初始解，再用这个初始解进行模拟退火。这样即可以让前一个时间片的模拟退火过程为后一个

时间片的初始解进行服务, 也可以在一定程度上利用时间片之间大量的公共信息, 并进行有效地转化, 减少公共资源的浪费。而即使在最糟糕的情况下, 也能与朴素模拟退火算法的求解效果相似。

首先, 如图 6 所示算法将提取上一个时间片的解矩阵  $L$ , 并找出其与当前可视矩阵  $V$  的公共部分, 然后在已有的子图  $L'$  上把可视矩阵  $V$  中具有而子图  $L'$  中没有的边按照花费从小到大在满足限制的情况下依次加入, 即可获得一个比较优秀的初始解。

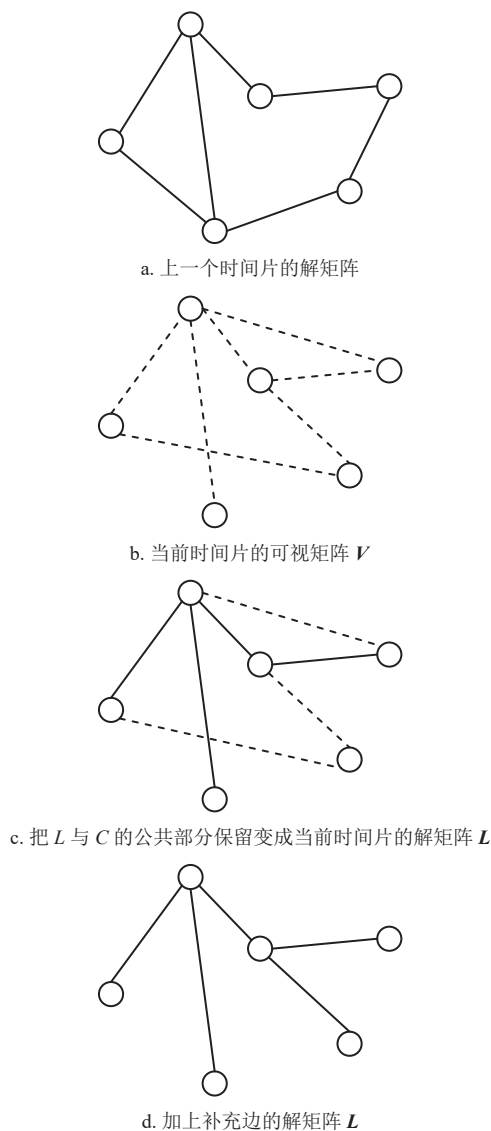


图 6 自适应生成初始解过程

提取上一个时间片的解矩阵  $L$  并把其与可视矩阵  $V$  的公共部分保留, 通过一次时间复杂度为  $O(n^2)$  的遍历, 将  $V$  中具有而  $L$  中没有的边进行排序, 整个过程的时间复杂度为  $O(n^2 \log n)$ 。

## 5 实验仿真与分析

本文假设在可视范围内的任意两颗卫星都可以存在相连的链路, 在给定网络拓扑和链路资源受限的情况下, 由于空间信息网络具有高动态性, 卫星节点间链路时时刻刻都在进行着断开与重连, 从而重构网络拓扑结构。本文选取了具有 66 颗卫星的 Iridium (铱星) 星座和 48 颗卫星的 Globalstar (全球星) 星座, 通过 STK 生成每个时间片内的可视矩阵数据。两个星座各具特点。铱星星座轨道为 6 个穿过两极上空的圆形轨道组成。每个轨道上有 11 颗均匀分布的卫星。通过这种形式可以完成全球覆盖。全球星星座则由 48 颗卫星组成, 其分布于 8 个倾角为  $52^\circ$  的圆形轨道上, 每个轨道 6 颗卫星。全球星星座所有的轨道都不经过极地。因此在非极地的部分, 卫星之间的可视性多有变化。但是由于全球星星座的卫星数量较少, 为了达成全球覆盖, 其卫星距地面高度约为铱星星座的两倍。这也导致星间的距离增大, 并且可视性的动态变化不如铱星星座频繁。

在仿真实验中, 每 60 s 划分一个时间片进行采样, 然后预处理求出每个时间片的数据。数据量越大, 实验结果的偶然性就会越小, 结果就会越准确, 但是如果选择时间片过多会导致模型求解缓慢, 所以选取了自 2021 年 1 月 1 日零时开始的 100 个时间片作为实验数据。本文中所提到的空间信息网络模型均满足:

$$\max \text{ edge} = 1.8n, \max \text{ degree} = 4 \quad (7)$$

设优化比例  $K$  为当前解与理论最优解的比例:

$$K = \frac{\tau_L}{\tau_{\min}} \quad (8)$$

该比例越接近 1, 就代表这个解越优秀, 优化的效果越好。但由于理论最优解是直接取为可视矩阵  $L$  的节点间平均最小时延, 且当前解相比于理论最优解增加了约束条件的限制, 因此该数值不可能等于 1。

实验将分别针对无边权模型和带边权模型, 对不进行迭代的初始解、贪心迭代得到的解、朴素模拟退火算法得到的解、自适应模拟退火算法得到的解进行比较。其中两种模拟退火算法使用的邻域求解算法都是双链路随机交换方式, 用以比较几种算法之间的计算差距。而后, 将会在同为朴素模拟退火算法和自适应模拟退火算法的情况下, 比较几种

不同的邻域求解方法的计算差异。

### 5.1 无边权模型实验结果

首先基于铱星星座得到的 4 种算法的平均优化比例。

从表 1 中可以得知, 没有进行迭代的初始解的优化比例最差, 而运用了贪心算法求出的解次之。而两种模拟退火算法在平均优化比例上仅存在十分细微的差距。值得注意的是, 从图 7 中可以看到, 在第 25 个时间片左右, 自适应模拟退火算法迭代出了一个相对较差的解。这是由于通过自适应方法选出的初始解不一定总是优秀的。由于空间信息的特性, 在相邻的时间片的变化通常十分细微, 但仍有一些个别的相邻时间片的动态变化会较大。这些变化较大的时间片在整个时间流程中占比很小 (大约只有不到 5%), 但是一旦其出现, 而仍然使用上一个时间片的信息构造新初始解, 则会出现模拟退火迭代出的解反而不如随机初始解的情况。而由于下一个时间片得到的解也是基于当前这个时间片, 这就会导致解的误差发生传递, 会让之后的所有时间片的解的误差不断扩大。因此不能盲目的使用上一个时间片的解信息。而应该综合考虑是否应该直接构造新解。

表 1 优化算法的平均优化比例及优化幅度 (铱星)

铱星星座	平均优化比例	优化幅度/%
初始解	1.603 754	0
贪心算法	1.496 612	6.68
朴素模拟退火算法	1.421 709	11.35
自适应模拟退火算法	1.421 020	11.39

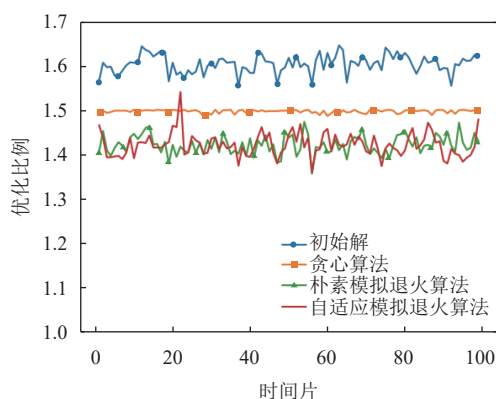


图 7 不同时间片下的优化比例 (铱星)

经过实验测算, 当确定了退出误差且相邻时间片差距较小时, 上一个时间片的迭代步数通常小于 100 步。因此, 只要在模拟退火中增加一个条

件: 当单次模拟退火的迭代步数已经超过 200 步, 且当前模拟退火的初始解是来自上一个时间片的解时, 则重新生成一个初始解, 然后再进行计算。这样计算的效率可以大大提升。

从表 2 中可以得知, 两种模拟退火算法的平均优化比例依旧十分相近。但值得注意的是, 贪心算法在全球星星座中也迭代到了一个相当不错的解。这是因为全球星星座的结构特殊, 其 8 条轨道排列的位置使得其对称性非常强, 再加上卫星的颗数更少, 导致其有效状态空间远少于铱星。

表 2 优化算法的平均优化比例及优化幅度 (全球星)

全球星星座	平均优化比例	优化幅度/%
初始解	1.934 234	0
贪心算法	1.651 279	14.63
朴素模拟退火算法	1.655 473	14.41
自适应模拟退火算法	1.645 647	14.92

还有一点值得注意的是, 自适应模拟退火算法在与时间片相关的图像中有好几段几乎水平的曲线, 这便是因为其继承上一个时间片的解的特性, 如果上一个时间片的计算误差小, 优秀的解被继承, 导致其在曲线的最后阶段得到了一个远优于另外两个算法的解。

由表 3 可以得知, 贪心算法的迭代步数远小于模拟退火算法, 但其运行时间也远大于模拟退火算法。这就说明贪心算法的迭代只有当能够得到一个比当前解更优秀的解才会进行迭代, 而模拟退火存在一定概率会接受到一个相对较差的中间解, 这就导致模拟退火的迭代步数会比贪心算法多。但是模拟退火单次迭代的复杂度为  $O(n^3)$ , 而贪心算法的单次迭代复杂度为  $O(n^6)$ , 这就导致贪心算法虽然迭代次数远少于两种模拟退火算法, 但是其运行时间反而远大于两种模拟退火算法。因此贪心算法实际上并没有两种模拟退火算法优秀。而两种模拟退火算法由于求邻域的方式相同, 所以其运行时间与迭代步数呈严格正相关。

下面对两种模拟退火算法直接进行对比分析: 随着时间片的不断推进, 两种模拟退火算法在两个星座下的运行时间和迭代步数如表 3 所示。首先可以发现, 由于求取邻域解的算法相同, 所以两种算法的运行时间与迭代步数呈现明显的正相关性。其次, 由于两个星座的各项参数不同, 在图中也可以发现算法的表现存在明显差异。铱星星座由于时间

片变化快, 所以自适应模拟退火算法并不具有太大的优势。而在变化不那么频繁的全球星星座, 自适应模拟退火算法中有多个曲线接近水平的曲线段, 这是因为其在这些位置继承了上一个时间片的信息, 并只需要几次迭代就可以得到一个优化比例相对较小的解。因此基本没有进行迭代就可以直接去求取下一个时间片的解。这就是自适应模拟退火算法的优势所在。

表 3 无边权模型下算法在铱星星座和全球星星座上的运行效率对比 (100 个时间片取和)

星座 算法	铱星星座 $\delta = 1.5$		全球星星座 $\delta = 1.7$	
	迭代步数	运行时间/s	迭代步数	运行时间/s
贪心算法	2 151	75.554	6 444	205.072
朴素模拟退火算法	67 409	12.010	86 693	6.409
自适应模拟退火算法	46 136	9.571	52 364	4.348

下面将分析 3 种不同的邻域求解算法在不同的星座上的表现。外层算法均使用自适应模拟退火算法, 只在邻域求解方法上做出差异。

由表 4 可以得知, 无论是在哪一个星座上, 生成树随机遍历算法都无法得到令人满意的结果。这是因为这种算法的随机程度不够高, 也没有确定的优化方向。通过 3 种邻域求解算法在 100 个时间片内的运行时间变化可以发现, 双链路随机交换方法和最大流选择方法在两个星座上表现各有优劣。全球星星座对称性强、有效状态少的特点使得随机性更强的双链路随机交换方法具有优势。而铱星星座变化快、卫星多、状态多的特点则更有利于最大流选择方法发挥其维护连通度的能力。未来的空间信息网络随着低轨卫星数量越来越多, 组网之后的网络拓扑结构将会日趋复杂, 最大流选择方法相较于双链路随机交换方法的优势也就会越来越大。

表 4 无边权模型下邻域求解算法在铱星星座和全球星星座上的运行效率对比 (100 个时间片取和)

星座 算法	铱星星座 $\delta = 1.5$		全球星星座 $\delta = 1.7$	
	迭代步数	运行时间/s	迭代步数	运行时间/s
生成树随机遍历	171 664	31.166	258 971	21.970
双链路随机交换	32 183	6.679	52 877	4.286
最大流选择	9 092	1.972	156 944	11.300

## 5.2 有边权模型实验结果

由无边权模型的分析可知贪心算法的效率极低, 因此不再分析。带边权模型中主要分析朴素模拟退火算法和自适应模拟退火算法的优劣, 以及

3 种邻域求解算法的优劣。

下面进行朴素模拟退火算法和自适应模拟退火算法的实验分析。同样的, 两种算法都使用双链路随机交换方法作为邻域求解方法。

由表 5 可知, 两种算法在迭代次数满的情况下对于两种星座的优化比例极其接近, 差距可以忽略不计。

表 5 有边权模型下算法的平均优化比例

算法	铱星星座	全球星星座
朴素模拟退火算法	1.035 067	1.154 427
自适应模拟退火算法	1.035 071	1.153 307

由表 6 和表 7 可知, 在铱星星座下, 自适应模拟退火算法的速度远快于朴素的模拟退火算法。而在全球星星座下, 虽然两种算法的迭代步数和运行时间大体相同。但是由于自适应模拟退火算法可以继承上一个时间片的优秀解, 加上全球星星座的变化较小, 所以其最终优化比例优于朴素模拟退火算法。

表 6 有边权模型下两种模拟退火算法在铱星星座下的运行效率 (100 个时间片取和,  $\delta = 1.037$ )

算法	迭代步数	运行时间/s	最终优化比例
朴素模拟退火算法	231 629	30.555	1.037 504
自适应模拟退火算法	13 414	2.096	1.036 185

表 7 有边权模型下两种模拟退火算法在全球星星座下的运行效率 (100 个时间片取和,  $\delta = 1.16$ )

算法	迭代步数	运行时间/s	最终优化比例
朴素模拟退火算法	211 477	11.299	1.170 666
自适应模拟退火算法	201 681	10.805 秒	1.162 185

在全球星星座中, 由于不同的时间片优化到某一优化比例的难度不同。而自适应方法通过之前的信息来求解, 更容易让这些求解简单的时间片直接得到优秀的解。

接下来将对 3 种不同的邻域求解方法进行比对。同样的, 外层的算法将使用带自适应的模拟退火算法。由于生成树随机遍历方法的运行效率低下, 因此主要比对剩下两种方法。

由表 8 中可以看出, 生成树随机遍历方法依然不够优秀。而剩下两种方法则各具优劣。其原因与无边权模型中的分析一致。

表 8 有边权模型下 3 种邻域求解算法的运行效率对比  
(100 个时间片取和)

星座 邻域求解	邻星星座 $\delta = 1.037$		全球星星座 $\delta = 1.16$	
	迭代步数	运行时间/s	迭代步数	运行时间/s
生成树随机遍历	295 100	41.771	271 508	15.781
双链路随机交换	11 648	1.892	186 659	9.864
最大流选择	7 725	1.464	236 105	14.672

## 6 结束语

本文针对复杂的空间信息网络进行了分析与建模,把核心问题聚焦在如何构建及优化空间信息网络拓扑结构使得卫星节点间通信传输时延平均值达到最小这一问题上。本文的主要工作集中在模拟退火算法的邻域解优化和自适应生成初始解两个方面。在邻域求解方面,提出了网络最大流选择方法邻域求解优化算法。在模拟退火算法初始解构造方面,提出了自适应生成时间片初始解。并在两种不同的且具有代表性的星座数据下对多种模型求解算法进行了实验验证。实验表明,在自适应生成初始解优化方面,模拟退火的计算效率明显提升。朴素的模拟退火算法和自适应模拟退火算法在优化比例上效果接近,但是在限定了退出误差的情况下自适应模拟退火算法的效率要优于朴素的模拟退火算法。本文提出的网络最大流选择方法作为邻域求解算法在两个具有代表性的星座构型下均优于生成树随机遍历方法,并与双链路随机选择方法互有优劣。当低轨卫星数量日益增多,网络拓扑结构不断发生改变,本文的空间信息网络优化模型的求解能有效的适用于组网策略上。

### 参考文献

- [1] 李德仁,沈欣,龚健雅,等.论我国空间信息网络的构建[J].武汉大学学报(信息科学版),2015,40(6):711-715.  
LI D R, SHEN X, GONG J Y et al. On the construction of my country's spatial information network[J]. Journal of Wuhan University: Information Science Edition, 2015, 40(6): 711-715.
- [2] 王彦鹏.卫星网络拓扑的抗毁性研究[D].大连:大连理工大学,2018.  
WANG Y P. Research on the survivability of satellite network topology[D]. Dalian: Dalian University of Technology, 2018.
- [3] BUTASH T, GARLAND P, EVANS B. Non-geostationary satellite orbit communications satellite constellations history[J]. International Journal of Satellite

- Communications and Networking, 2021, 39(1): 1-5.
- [4] CHEN Q, GIAMBENE G, YANG L, et al. Analysis of inter-satellite link paths for LEO mega-constellation networks[J]. IEEE Transactions on Vehicular Technology, 2021, 70(3): 2743-2755.
- [5] FRANK H, FRISCH I T. Analysis and design of survivable network[J]. IEEE Transactions on Communication Technology, 1970, 18(5): 567-662.
- [6] BOESCH F T, THOMAS R E. On graphs of invulnerable communication net[J]. IEEE Transactions on Circuit Theory, 1970, 17(2): 183-192.
- [7] 向尚,陈盈果,李国梁,等.卫星自主与协同任务调度规划综述[J].自动化学报,2019,45(2):252-264.  
XIANG S, CHEN Y G, LI G L, et al. Review on satellite autonomous and collaborative task scheduling planning[J]. Acta Automatic Sinica, 2019, 45(2): 252-264.
- [8] 燕洪成,张庆君,孙勇.星间链路数量受限的导航卫星网络链路分配问题[J].航空学报,2015,36(7):2329-2339.  
YAN H C, ZHANG Q J, SUN Y. Link assignment problem of navigation satellite networks with limited number of inter-satellite links[J]. Acta aeronautica et Astronautica Sinica, 2015, 36(7): 2329-2339.
- [9] 石磊玉,向为,唐小妹.一种兼顾卫星导航系统星间观测及通信的链路分配算法[J].宇航学报,2011,32(9):1978-1985.  
SHI L Y, XIANG W, TANG X M. A link assignment algorithm applicable to crosslink ranging and data exchange for satellite navigation system[J]. Journal of Astronautics, 2011, 32(9): 1978-1985.
- [10] 董明佳,林宝军,刘迎春,等.基于多目标模拟退火算法的导航卫星激光星间链路拓扑动态优化[J].中国激光,2018,45(7):217-228.  
DONG M J, LIN M J, LIU Y C, et al. Topology dynamic optimization for inter-satellite laser links of navigation satellite based on multi-objective simulated annealing method[J]. Chinese Journal of Lasers, 2018, 45(7): 217-228.
- [11] 潘成胜,行贵轩,戚耀文,等.多状态空间信息网络拓扑生成优化算法[J].航空学报,2020,41(4):221-230.  
PAN C S, XING G X, QI Y W, et al. Topological generation and optimization method in multi-state space information network[J]. Acta aeronautica et Astronautica Sinica, 2020, 41(4): 221-230.
- [12] 宋炜琳,杨道宁.基于星间链路的卫星导航系统星地业务信息传输规划调度方法研究[J].兵工学报,2019,40(8):1627-1633.  
SONG W L, YANG D N. Research on GNSS satellite-ground service information transmission scheduling method based on inter-satellite link[J]. Acta

- Armamentarii, 2019, 40(8): 1627-1633.
- [13] 周雅, 谢卓辰, 刘沛龙, 等. 基于区域分流的低轨卫星星座星间负载均衡路由算法[J]. 中国科学院大学学报, 2021, 38(5): 687-695.
- ZHOU Y, XIE Z C, LIU P L, et al. Inter-satellite load balancing routing algorithm for LEO satellite constellation based on regional-traffic-detour[J]. Journal of University of Chinese Academy of Sciences, 2021, 38(5): 687-695.
- [14] 董飞鸿, 吕晶, 巩向武, 等. 空间信息网络结构抗毁性优化设计[J]. 通信学报, 2014, 35(10): 50-58.
- DONG F H, LYU J, GONG X W, et al. Optimization design of structure invulnerability in space information network[J]. Journal on Communications, 2014, 35(10): 50-58.
- [15] ZHUO M, YANG P, CHEN J Y, et al. Adaptive optimization of dynamic heterogeneous network topologies a simulated annealing methodology[J]. Lecture Notes in Computer Science, 2022, 13339: 587-612.
- [16] YANG P, ZHUO M, TIAN Z W, et al. Optimization of space information network topology based on spanning tree algorithm[J]. Communications in Computer and Information Science 2022, 1587: 668-679.
- [17] WAN S M, YANG P, ZHUO M, et al. An improved space information networks topology algorithm[C]//2021 IEEE International Conference on Information Communication and Software Engineering. [S.l.]: IEEE, 2021: 227-232.
- [18] 托马斯·科尔曼, 查尔斯·雷瑟尔森, 罗纳德·李维斯特, 等. 算法导论[M]. 潘金贵, 顾铁成, 译. 北京: 机械工业出版社, 2006.
- THOMAS H C, CHARLES E L, RONALD L R, et al. Introduction to algorithms[M]. Translated by PAN J G, GU T C, Beijing: China Machine Press, 2006.

编辑 税红