



北京航空航天大学  
BEIHANG UNIVERSITY

# 硕士学位论文 中期报告

## 基于 SDN 的低轨卫星网络控制器动态 部署研究

学    院：电子信息工程学院  
专    业：交通信息工程及控制  
姓    名：张家宁  
学    号：ZY1902307  
指导教师：张  涛

2021 年 7 月

## 目录

|                                 |    |
|---------------------------------|----|
| 第一章 课题来源及意义.....                | 2  |
| 1.1. 课题背景.....                  | 2  |
| 1.2. 国内外研究现状.....               | 3  |
| 1.2.1. 控制器部署研究现状.....           | 3  |
| 1.2.2. SDN 路由算法研究现状.....        | 3  |
| 1.2.3. SDN 动态网络拓扑 CPP 研究现状..... | 4  |
| 1.3. 研究目的和意义.....               | 4  |
| 第二章 研究内容.....                   | 5  |
| 2.1. 课题研究内容.....                | 5  |
| 2.1.1. SDN 静态卫星拓扑控制器部署研究.....   | 5  |
| 2.1.2. SDN 动态卫星拓扑控制器部署研究.....   | 9  |
| 2.1.3. SDN 卫星网络仿真平台.....        | 13 |
| 第三章 关键技术及难点.....                | 16 |
| 第四章 完成情况.....                   | 17 |
| 第五章 预期目标和进度安排.....              | 17 |
| 5.1. 进度安排.....                  | 17 |
| 第六章 参考文献.....                   | 17 |

# 第一章 课题来源及意义

## 1.1. 课题背景

随着 5G 标准的初步完善和商用部署的逐步推进，移动网络的发展全面进入了一个新时代，5G 网络逐步在 eMBB（增强移动带宽）、mMTC（海量机器类通信）、URLLC（高可靠低时延通信）三大应用场景的成熟应用标志着移动通信网络在网络延迟、数据速率、移动性和连接数量方面都取得了长足的进步，但是，随着科技的进步和社会的发展，人们对智能化的需求不断提升，5G 网络覆盖不足的缺陷无法满足人们对于随时随地通信的需求，因此 6G 网络研究的浪潮开始在各国掀起，万网互联成为 6G 网络研究的一大重点，万网互联主要涉及的应用场景是天地一体化融合网络，该场景利用相互连接的低地球轨道卫星和其他的非地球网络节点和平台，构建一个与地面网并行的空间网络，通过空间与地面网络融合实现包括海洋、沙漠、边远乡村、以及船只和飞机等移动平台在内的无处不在的互联网接入，满足人们对通信广覆盖低延迟的通信需求。

天地一体化融合网络的核心是低轨卫星网络，低轨卫星网络由于其全球覆盖与较低的轨道高度，低轨道卫星星座具有传输损耗小、通信时延短、功耗低等优点，可以提供低延迟的通信能力。然而，时变的卫星网络拓扑给网络管理与鲁棒性路由算法的设计带来了巨大的挑战，传统卫星网络的分布式路由算法收敛速度慢、网络配置繁琐、路由策略不灵活，星地距离长使卫星与地面站一对一的控制结构在更新时耗时大，所以当前的卫星网络不能适应网络状态的不断变化和服务的精细化，此外系统间的互联更加困难。

软件定义网络（Software Defined Network, SDN）代表了通信网络向逻辑集中式架构的范式转变，它将数据平面和控制平面分离，这种解耦使得网络可以通过各种应用程序直接编程，例如北行接口上的 REST 和南行接口上的 OpenFlow。这种模式简化了网络管理，提高资源利用率。将 SDN 应用在低轨道卫星网形成软件定义卫星网，与传统卫星网相比，软件定义卫星网有着平衡空间动态路由策略的灵活性与可控性、快速部署和更新卫星网络策略、提供灵活、精细、可伸缩的网络控制、提高卫星之间的协作与异构空间系统的兼容性等优势。

SDN 最关键的问题就是控制器部署问题（Controller Placement Problem, CPP），这将影响从延迟到弹性、从能效到负载平衡等一系列网络问题。近年来关于 SDN 的 CPP 的研究层出不穷，但几乎都是针对静态的地面网络进行研究的，与地面网络相比，低轨道卫星网络具有拓扑高动态、业务动态分布、链路时延长、传输可靠性低、星上资源有限等特点，这使得用于解决地面网络的 SDN 部署问题的方法不再适用。本文针对软件定义卫星网络中的控制器部署问题，关注网络延

迟,提出一种适用于低轨卫星网络的控制器动态部署方案,使得网络在满足低延迟的情况下尽量实现控制器的负载平衡,最后通过仿真探究该方案的可行性与对网络性能的改善。

## 1.2. 国内外研究现状

### 1.2.1. 控制器部署研究现状

Heller 等人[1]是第一篇在 SDN 中介绍和制定 CPP 的论文。他们提出了三个衡量标准,即,第一个是对于给定数量的控制器,最小化平均情况下的开关控制器等待时间,这可以映射到  $k$ -中值问题,第二个是对于给定数量的控制器,最小化最坏情况下的开关控制器等待时间,类似于  $k$ -中心问题,以及第三个是在等待时间界限内最大化每个控制器的开关数量,如集合覆盖问题。他们将其分析限制在传播延迟最重要的广域网上,并根据互联网拓扑评估控制器数量和交换机控制器传播延迟之间的权衡。这篇论文提供了一个对 CPP 的透彻的见解,这已经被深入研究。从传播延迟的角度来看,作者得出结论,一个控制器通常足以满足大多数广域网(尽管需要多个控制器来满足容错和弹性要求)。

Obadia 等人[2]研究了 CPP,以减少 SDN 控制开销。作者将控制平面开销建模为统计收集成本(包括交换机控制器延迟)、流设置成本(包括初始路径设置请求、中间路径设置请求和规则安装成本)和同步成本(包括控制器间延迟)的总和。作者不是将控制器间通信建模为所有控制器对之间的延迟之和,而是根据控制器的生成树对其进行建模。一个 MILP 公式被用来推导最优控制器布局以及控制器之间的最小生成树。还提出了一种贪婪启发式算法。该方法在真实拓扑和随机拓扑上得到了验证,并且发现贪婪启发式算法能够提供接近最优的结果。

Hu 等人[3]将交换机-控制器路径上的链路数量作为度量来处理能量感知的 CPP。他们提出了一个二进制整数规划(BIP)以及一个基于遗传算法的启发式算法。将结果与 SND 拓扑上的电容受限控制器布局问题进行了比较。节能是以增加交换机控制器延迟为代价的。与 BIP 相比,该启发式算法要求不超过 4%的额外链路,假设每个链路的能耗相同。

### 1.2.2. SDN 路由算法研究现状

在[4]中,设计了一种针对卫星网络的容错路由机制,设计了一种基于故障块的容错路由算法。将卫星节点和消息进行分类,使数据包能够绕开故障传输,从而达到容错的目的。

在[5]中, 针对海量低轨道卫星交换机, 考虑存在某些路径必须通过某些卫星节点情况下的路由问题, 提出了将深度优先搜索(DFS)思想融入 Dijkstra 的算法, 来解决海量交换机的路由问题, 提高了计算的效率与可靠性。

在[6]中, 提出了一种简单的最短路径算法以及拥塞避免算法, 算法十分简单资源消耗小, 但是求得的路径并不是最优。

### 1.2.3. SDN 动态网络拓扑 CPP 研究现状

在[7]中, 提出了 OpenSAN 软件定义卫星网络架构, 包含低轨卫星网络、同步轨道卫星、地面站与地面网络操作控制中心, 低轨卫星网络作为数据平面, 同步卫星与地面网络操作控制中心分别作为域控制器与主控制器, 通过分层的控制器结构, 解决了控制器的部署问题, 有理论可行性, 但是由于同步卫星通信延迟过长, 必然会使得网络的流响应时间很大, 影响网络性能, 且提出的架构需要地面低轨卫星与同步卫星, 架构复杂度高, 无法应对链路或者节点失效的情况。

在[8]中改进了 OpenSAN, 将地面站与某些低轨卫星链接, 增加了控制器与低轨卫星的通信方式, 降低了同步轨道卫星链路带来的高响应时间, 但是此方案的改进程度与地面站的数量严格相关, 即需要大量地面站才能改善网络的流响应时间。

在[9]中, 提出了一种低轨道星载控制器策略 SoftLEO, 使用永久性的星间链路连接控制器与交换机, 采用水平控制结构, 在每个卫星轨道上选择编号相同的卫星作为该轨道的控制器, 通过轨道间星间链路连接控制器, 形成逻辑上的集中控制, 该方案为了避免了星间链路变化带来的问题, 完全去掉了对星间链路的依赖, 采用固定控制器的方式, 这种方式虽然简单, 但是固定的控制器位置必然会因为流量的动态分布造成流响应时间增加。

由于卫星流量需求的动态性与网络拓扑的动态性, 固定位置的控制器很难满足低流响应时间的需求, 在[10]中, 提出了动态的控制器部署方法, 考虑了用户流量的动态性与卫星拓扑的动态性, 针对流量随地理位置与时间变化的特征, 需求提出了动态控制器部署问题, 并将该问题建模为一个整数线性规划问题, 目标函数是使平均流响应时间最小, 求解目标与流量位置相关联, 更加合理。但是该方法采用卫星快照的方式使得星上存储压力较大, 快照的方式也无法检测到链路或者卫星故障, 同时文章没有讨论动态迁移控制器所带来的问题。

## 1.3. 研究目的和意义

由于 SDN 中数据和控制平面的内在去耦, SDN 交换机向 SDN 控制器查询(使用分组输入消息)它遇到的每个新流的适当转发动作。因此, SDN 中的流建立时间

取决于交换机-控制器通信延迟。所以，SDN 控制器的放置是一个关键的设计考虑因素，因为它直接影响 SDN 交换机经历的控制延迟，进而影响广泛的网络问题，如路由策略更新、负载平衡、弹性、故障管理、服务质量等。尽管研究表明，大多数网络只需要一个控制器就能满足网络通信的需求，但这会带来单点故障问题和高延迟问题，需要多个控制器来提高网络的弹性。

## 第二章 研究内容

### 2.1. 课题研究内容

本文研究的问题是低轨道卫星网络下的多目标优化的 SDN 控制器部署问题，优化的目标主要有两个：交换机-控制器延迟和控制器负载平衡。研究内容主要有三个：SDN 静态卫星拓扑控制器部署研究、SDN 动态卫星拓扑控制器部署研究、SDN 卫星网络仿真平台。

#### 2.1.1.SDN 静态卫星拓扑控制器部署研究

控制器部署问题 (Controller Placement Problem, CPP)，由于 SDN 中数据和控制平面的内在去耦，SDN 交换机向 SDN 控制器查询(使用分组输入消息)它遇到的每个新流的适当转发动作。因此，SDN 中的流建立时间取决于交换机-控制器通信延迟。所以，SDN 控制器的放置是一个必须考虑的关键的因素，因为它直接影响 SDN 交换机经历的控制延迟，进而影响广泛的网络问题，如路由策略更新、负载平衡、可扩展性、能效、弹性、故障管理、服务质量等。

CPP 的几个方面值得关注。可以从网络响应性、容错性、弹性、服务质量、负载平衡等不同目标的角度来审视 CPP。虽然大多数文献中对单个目标的研究最多，但通常需要考虑多个相互冲突的目标，包括它们之间的权衡。CPP 的另一个方面的研究也相当热门，就是使用图论、博弈论、排队论等概念对公式化的 CPP 建模的方法。以及潜在的建模假设。众所周知的设施位置问题[2]可简化为 SDN [3] 中的 CPP (控制器作为设施，开关作为需求点)，从而建立后者的 NP-hardy。SDN 中的 CPP 也可以从顶点覆盖问题[4]或从支配集问题[5]进行约简。因此，用于获得 CPP 解决方案的方法也涵盖了很广的范围。已经研究了多种策略——最优策略和启发式策略——以获得合适的 CPP 解决方案。

CPP 目前比较新的研究方向是 CPP 的应用领域研究。随着 SDN 在数据中心、广域网、移动网络、5G 等广泛领域获得发展势头。CPP 在新兴的应用领域进行探

索,不同领域的不同特征使得 CPP 需要针对这些特征为不同领域专门制定解决方案。

卫星网络的控制部署问题主要解决两个问题:一个是确定卫星网络中需要的控制器数量,另一个是确定这些控制器布置在网络中的位置。本文将这部分内容分为三块来研究,分别是构建卫星拓扑模型、确定卫星拓扑控制器数量,确定控制器位置。

本文首先对控制器的部署问题进行建模,建立了以下基本模型:

**网络模型:** 以无向图  $G(V, E, C)$  表示一个网络图,其中  $V$  是交换机的集合,因此一个网络图总共有  $2^V$  种布局,  $E$  是连接这些交换机的链路,  $C$  是控制器的集合,并且  $c_i$  与  $v_i$  表示交换机和控制器位于同一位置。以  $d(u, v)$  表示交换机  $u, v \in V$  之间最短路径上的延迟。  $P$  表示控制的容量,并假设所有控制器的容量都一样。以  $\lambda_{v_i}$  表示交换机  $v_i$  到其所属控制的负载,并假设  $\lambda_{v_i} = 1, v_i \in V$ 。以  $\varphi(v)$  表示  $V \rightarrow C$  的映射关系。最后以  $\Delta$  表示交换机到控制器的延迟阈值。

**CPP 公式:** 本文以以下映射关系表示 CPP 问题中控制器位置和交换机与控制器的对应关系,如公式(1)所示:

$$\begin{aligned} x_i &= \begin{cases} 1, c_i \text{ 存在} \\ 0, \text{其他} \end{cases} \\ y_{ij} &= \begin{cases} 1, v_i \text{ 映射到 } c_j \\ 0, \text{其他} \end{cases} \end{aligned} \quad (1)$$

这里决策变量  $x_i$  表示控制器是否存在,  $y_{ij}$  表示交换机  $v_i$  属于控制器  $c_j$ 。

本文把控制器部署问题看作是一个优化问题,优化目标以及相关约束条件如下列公式所示:

优化目标:

$$\min_{2^V} \max_{v \in V} \min_{c \in C} d(v, c) \quad (2)$$

约束方程:

$$\forall c_j \in C: \sum_{v_i \in V} \lambda_{v_i} y_{ij} \leq \rho x_i \quad (3)$$

$$\forall v_i \in V: \sum_{c_j \in C} y_{ij} = 1 \quad (4)$$

$$\forall v_i, c_j \in V: y_{ij} \leq x_i \quad (5)$$

$$\forall v_i, c_j \in V : d(v_i, v_j) y_{ij} \leq \Delta \quad (6)$$

$$\forall v_i, c_j \in V, x_i, y_{ij} \in \{0,1\} \quad (7)$$

距离公式：

$$\begin{cases} dl(v1, v2) = 2R_e \sqrt{\sin^2(lat1 - lat2)/2 + \sin^2(lon1 - lon2)/2 \bullet \cos(lat1) \cos(lat2)} \\ d(v1, v2) = \sqrt{dl^2(v1, v2) + (h1 - h2)^2} \end{cases} \quad (8)$$

本文对控制器部署问题的优化目标是最小化最大交换机-控制器延迟，如公式（2）所示，相关的约束方程列在（3）-（7）中。（3）为控制器容量约束方程，表示接入到控制器的负载不超过其容量。约束方程（4）表示每个交换机只映射到一个控制器上，方程（5）表示只有在控制器  $x_j$  存在的情况下，交换机  $v_i$  才能映射到  $x_j$ 。方程（6）表示交换机-控制器延迟不超过延迟阈值。方程（7）表示决策变量满足二元性。

建立模型后，在某一时间片的开始时刻，认为卫星网络拓扑不会发生变化，先使用基于球坐标系的聚类算法将卫星网按节点之间的距离分为  $k$  类，每个类分配一个控制器，即确定控制器数量为  $k$ ，接下来需要确定这些控制器需要放在网络中的哪个位置，针对这个问题，选出使得最大交换机-控制器时延最小的部署方法为所求，这样就能确定某一时间片内卫星网的控制器部署情况了。静态卫星拓扑控制器部署流程图如图 1 所示，图 2 为选定  $k$  的最佳值，图 3 为聚类收敛情况，图 4 为聚类结果结果：



图 1：静态卫星拓扑控制器部署流程图



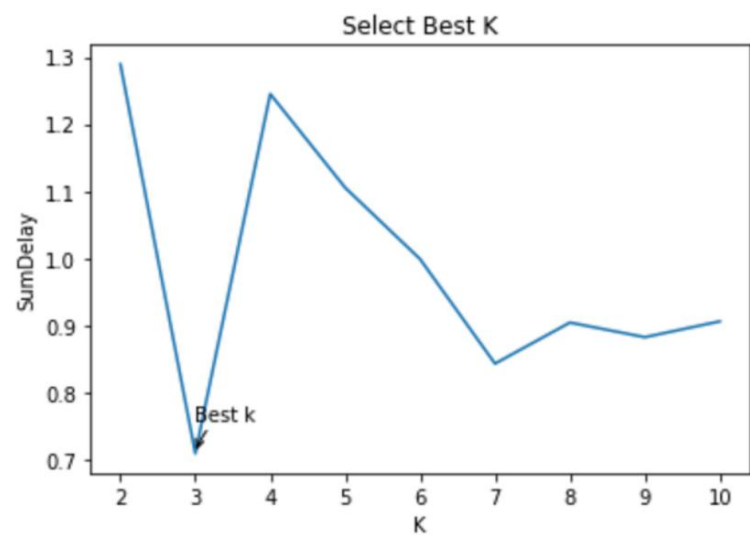


图 2：选择最佳 k 值

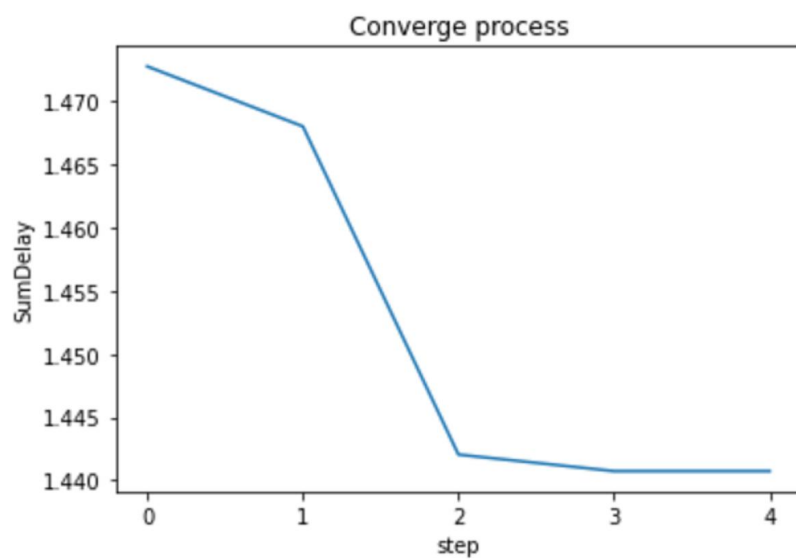


图 3：聚类收敛过程

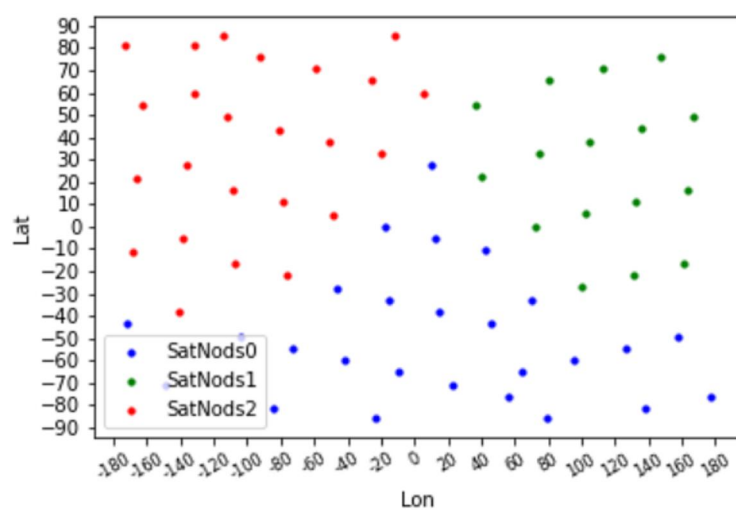


图 4：铱星拓扑聚类结果二维图

## 2.1.2.SDN 动态卫星拓扑控制器部署研究

这部分研究分为两个部分展开：控制器交换机控制关系切换和控制关系切换时控制更新控制信息。

由于卫星网络动态变化的特性，在一个时间点确认了控制器和交换机的控制关系后，在下一个时间点这个控制关系将不再适用，因此需要实时切换控制器和交换机之间的控制关系并且在控制关系切换时，各个控制器需要交换各自的控制信息来更新控制信息。

基于这个卫星网的动态特性，本文提出了适用于 SDN 动态卫星拓扑的星地双层架构，示意图如图 5 所示：

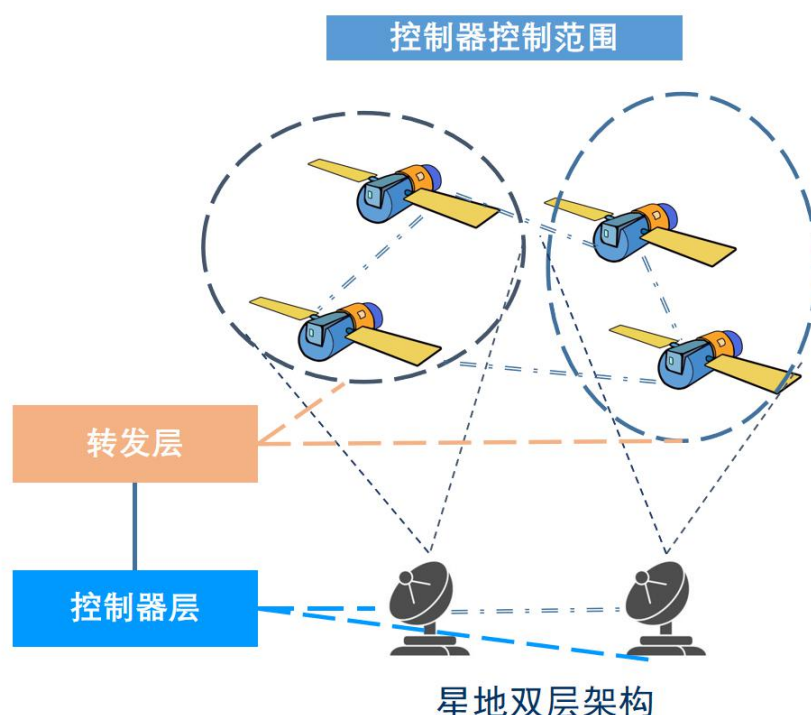


图 5：星地双层架构

将 SDN 卫星网分为两层：转发层和控制层，其中转发层由卫星节点构成，卫星只负责根据流表信息转发数据包，控制器层为地面层，将控制器部署在地面，对初始卫星拓扑进行静态网控制器部署，确定地面控制器数量、位置以及控制范围，地面控制器动态监测进入其控制范围内的卫星并建立控制映射关系，负责向卫星层发送流表，当控制映射关系改变时，控制器之间交互控制信息。

在对初始化卫星拓扑进行静态控制器部署后，确定了地面控制器的数量和位置信息（经纬度），以及各个控制器的控制范围，并且这也是整个架构的控制器最终位置和数量，因为根据卫星网的运动特性，在任一空间区域内，卫星的数量是均匀的，在此证明一下卫星的空间均匀性：

### 2.1.2.1. 卫星的空间均匀性

论文[11]证明了在均匀分割为一个个小区域的地面上，每个小区出现的星下

点的次数相同，在此基础上，证明了卫星的空间均匀性。

设：有  $N$  个卫星，地球是一个理想的球，卫星绕地球做圆周运动，把地球分为一个个面积为  $\Delta s$  的区域，每个控制器的控制范围  $d_i$  为半径为  $R_i$  的圆，其在地面的投影面积可以由  $\Delta s$  积分获得。总的  $M$  个控制区域，

$\tau^T(\Delta s)$ ：表示  $T$  时间内，卫星下点位于  $\Delta s$  区域的总时间

$\tau_i^T(d_i)$ ：表示  $T$  时间内，卫星位于  $d_i$  区域的总时间

$\tau_i^T(d_i, S_k)$ ：表示卫星  $S_k$  位于区域  $d_i$  的总时间

$\tau^T(\Delta s, S_k)$ ：表示卫星  $S_k$  星下点位于  $\Delta s$  区域的总时间

$T_k$ ：卫星  $S_k$  运动的总时间

则有如下关系：

$$\tau_i^T(d_i) = \sum_{k=1}^N \tau_i^T(d_i, S_k) = \sum_{k=1}^N R_i \tau^T(\Delta s, S_k) \quad (9)$$

$$T_k = \sum_{j=1}^M \tau_j^T(d_j, S_k) = \sum_{j=1}^M R_j \tau^T(\Delta s, S_k) \quad (10)$$

由论文[11]中的结论可以导出：对于任意两个不同的区域  $m_1$ 、 $m_2$ ：

$$\lim_{T \rightarrow \infty} \frac{\tau_{m_1}^T(d_{m_1})}{\tau_{m_2}^T(d_{m_2})} = \lim_{T \rightarrow \infty} \frac{\sum_{k=1}^N \tau_{m_1}^T(d_{m_1}, S_k)}{\sum_{k=1}^N \tau_{m_2}^T(d_{m_2}, S_k)} = \lim_{T \rightarrow \infty} \frac{\sum_{k=1}^N R_{m_1} \tau^T(\Delta s, S_k)}{\sum_{k=1}^N R_{m_2} \tau^T(\Delta s, S_k)} = \frac{R_{m_1}}{R_{m_2}} \quad (11)$$

即在足够长的时间里，不同控制器的控制区域内出现的卫星数量与控制区域的半径成正比，设  $R_s$  为卫星轨道面总面积的半径，则在一段时间内，控制器控制区域  $m$  内的卫星数量均匀，为  $NR_m/R_s$ 。

在确定了控制器数量和位置后，控制器只需要在卫星网动态变化时确定在其控制范围内的卫星节点，并与其建立控制关系即可。

下面通过仿真来验证卫星网的空间均匀性，仿真流程如图 6 所示，仿真结果如图 7 所示：

每隔一分钟，从卫星仿真平台获取铱星系统所有卫星在一个周期内的位置信息，共100帧卫星拓扑信息

对第一帧拓扑使用静态拓扑部署策略后，确定控制位置和数量，以及控制范围。

对每一帧使用动态拓扑控制器部署策略，并统计每个控制器的控制范围内的卫星数量

将结果与理论推导比较，结论是否正确

图 6：验证空间均匀性流程图

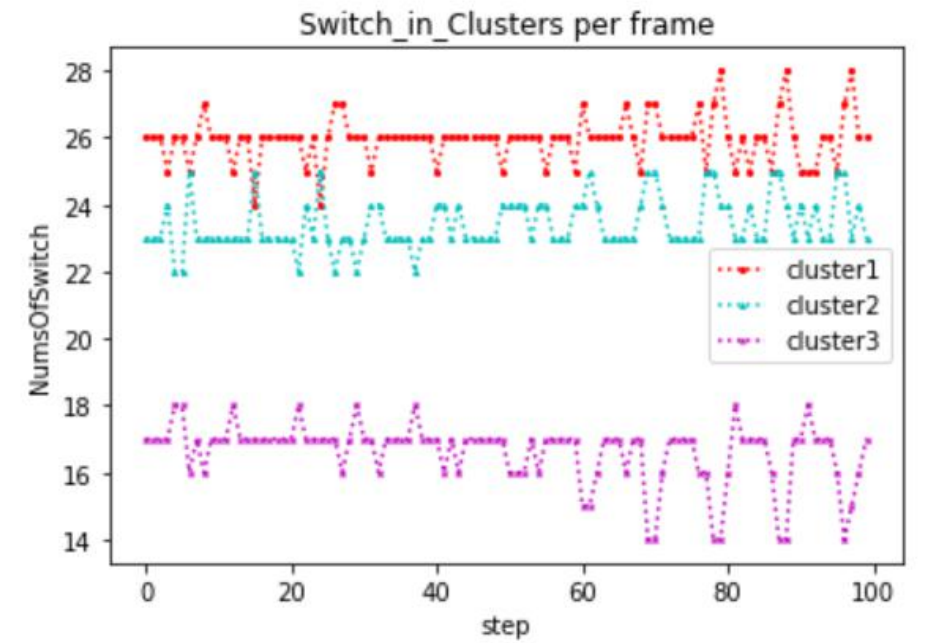


图 7：验证空间均匀性仿真图

由仿真结果可知，各个控制器控制范围内的卫星数量在每一帧时基本保持不变，最大波动为 3，因此卫星的空间均匀性得到验证。

#### 2.1.2.2. 动态控制器部署策略验证

SDN 卫星网动态控制器部署策略流程图如图 8 所示，仿真验证如图 9 所示：

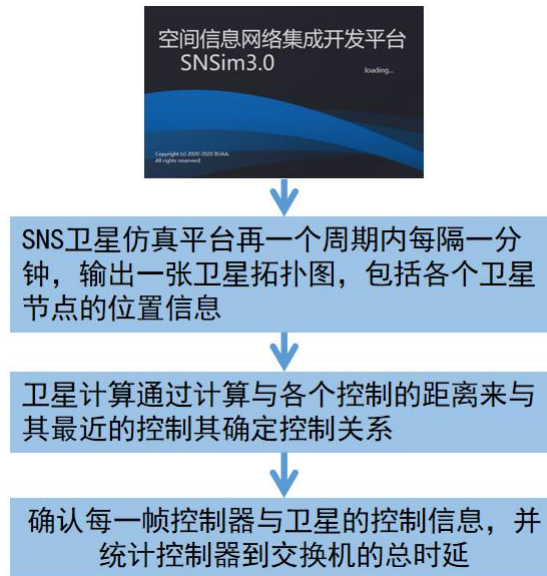


图 8: 动态控制器部署策略流程图

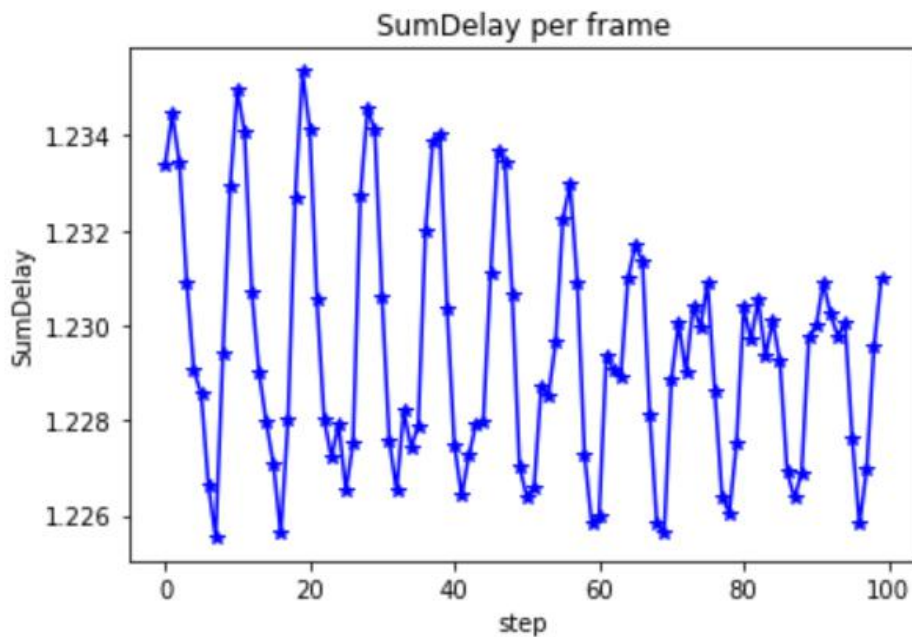


图 9: 动态控制器部署策略仿真图

由仿真结果可知，每一帧的总延迟为 1.23s，最大波动为 8ms，每一帧的总延迟稳定。

### 2.1.2.3. 控制器更新控制信息

地面控制器负责向受其控制的卫星交换机发送流表，当卫星拓扑变化时，地面控制器和卫星交换机的控制关系改变，若老控制器还有待下发给控制的交换机的流表时，就需要将这些下发信息转交给新控制器。本文提出了一种控制器更新

控制信息的策略，如图 10 所示：

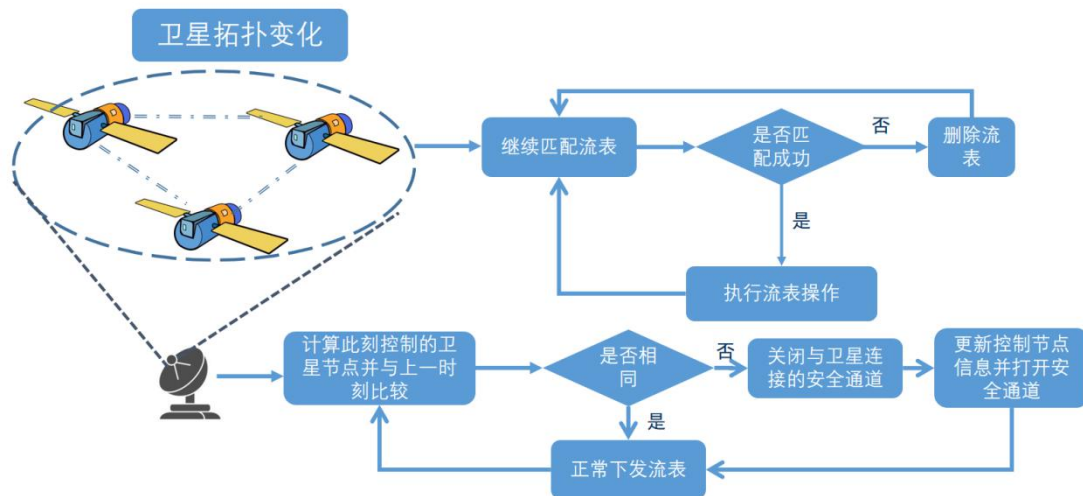


图 10: 控制器更新控制信息流程图

### 2.1.3.SDN 卫星网络仿真平台

在提出了 SDN 卫星动态控制部署策略后，需要搭建仿真平台设计实验来验证所提出方法的可行性、正确性并客观地衡量方法的优缺点。

为了验证上述方案，需要在仿真平台上进行仿真，本文拟采用 mininet 与 ryu 进行仿真，mininet 是一个开源 SDN 实时仿真软件，它集成了 docker 与 open vswitch，可以使用 docker 在消耗极小资源的情况下虚拟出多个网络相互独立的虚拟主机，open vswitch 通过软件实现支持 openflow 的交换机，ryu 则是一款简单的 SDN 控制器。

仿真的主要工作有：在 docker 虚拟机中模拟地面动态流量情况，并生成用户流量，扮演地面用户角色；使用 open vswitch 代表低轨道卫星交换机；ryu 作为控制器，除了基本的控制配置模块外还需要获取网络状态，并进行控制器部署以及实现动态网络下的控制器迁移。

本文设计的实验主要是对比实验：针对所设计的动态部署算法，都会与时间片放法求解的控制器部署方法作比较，以评估所设计的方案的可行性与有效性。

#### 2.1.3.1. 卫星星座模型

本文采用铱星系统进行仿真验证，铱星系统于 1990 年正式立项，1996 年开始发射卫星并于 1998 年提供商业服务。它采用近极地的低轨道（LEO Low Earth Orbit Satellite）星座，所有卫星的轨道高度均为 780km，各个轨道的倾角为 86.4°。



经优化后，整个铱星系统的卫星数量为 66 颗，它们被均匀分布在 6 个轨道面上。图 11 和图 12 分别为铱星的三维和二维示意图。

Iridium 系统的主要特点如下：（1）实现包括南北两极区域在内的全球无缝覆盖，；（2）卫星间采用全双工的星间链路来通信；（3）卫星具有星上处理和交换功能。这些特点在当时极为先进，但同时也增加了复杂性，提高了运营成本。铱星系统除了在全球范围提供蜂窝电话业务外，还提供传真、全球定位（GPS）、无线电定位，以及全球寻呼业务。目前主要为美国军方提供服务，用于军事通信、应急搜救等方面。

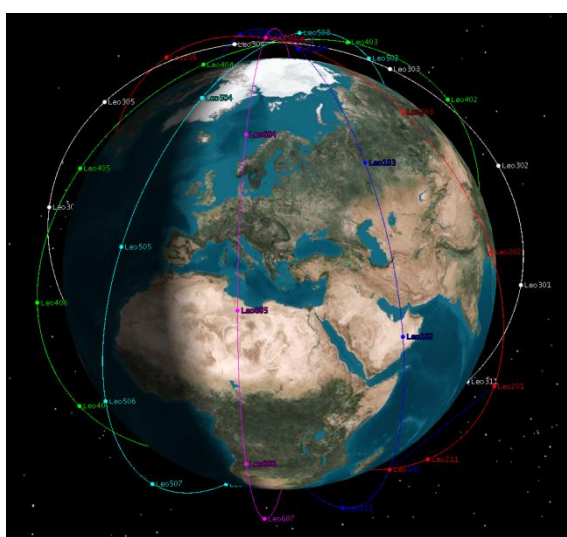


图 11：铱星星座三维图

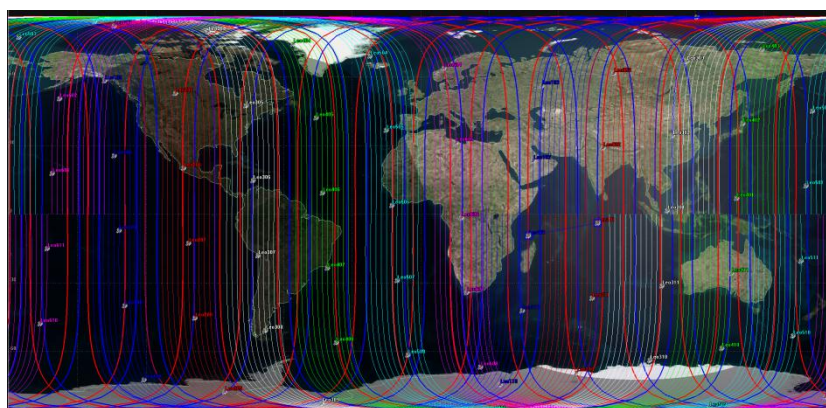


图 12：铱星星座二维图

### 2.1.3.2. 星座仿真流程

卫星星座仿真平台主要由三个模块组成，分别是卫星模型生成模块，卫星拓扑处理模块

以及 SDN 仿真模块。卫星模型生成模块，卫星拓扑处理模块的功能如图 13、图 14 所示：

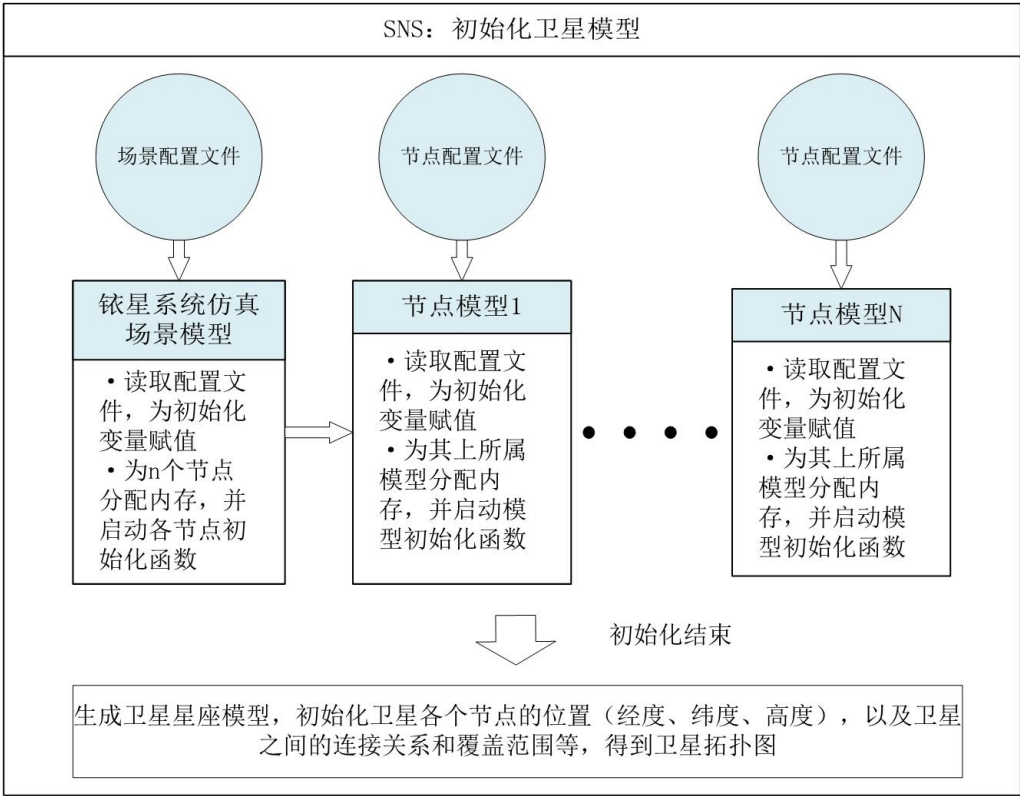


图 13：卫星模型生成模块

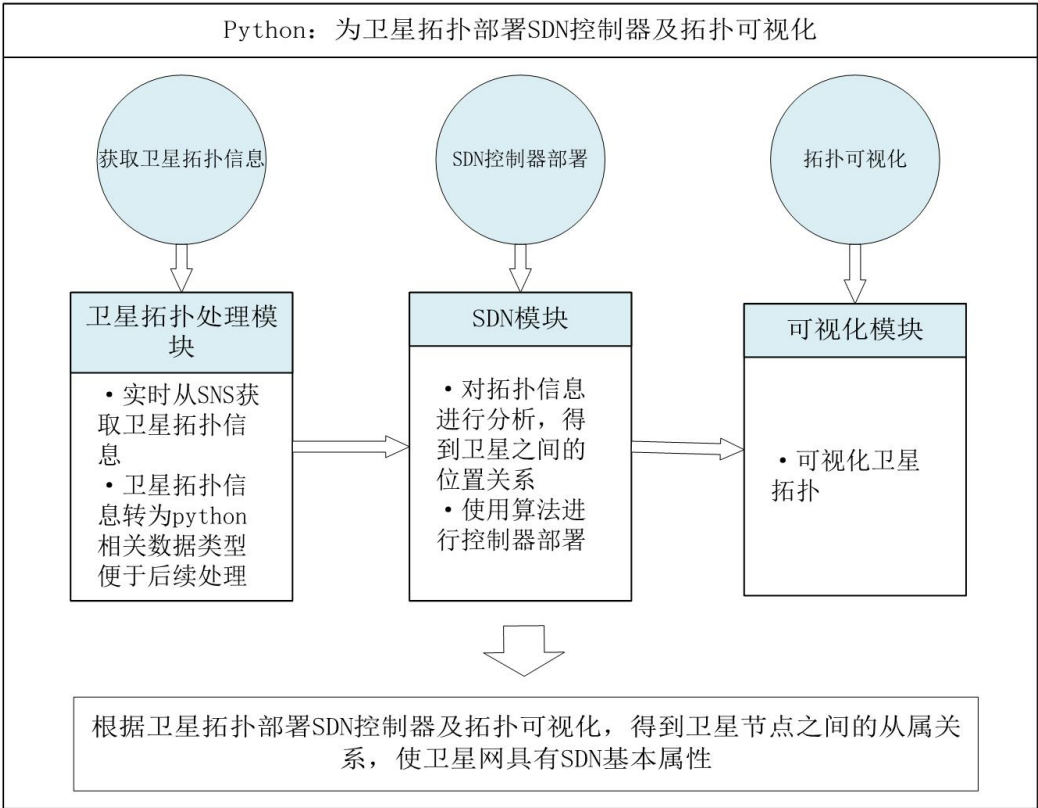


图 14：卫星拓扑处理模块



仿真流程图如图 15 所示：

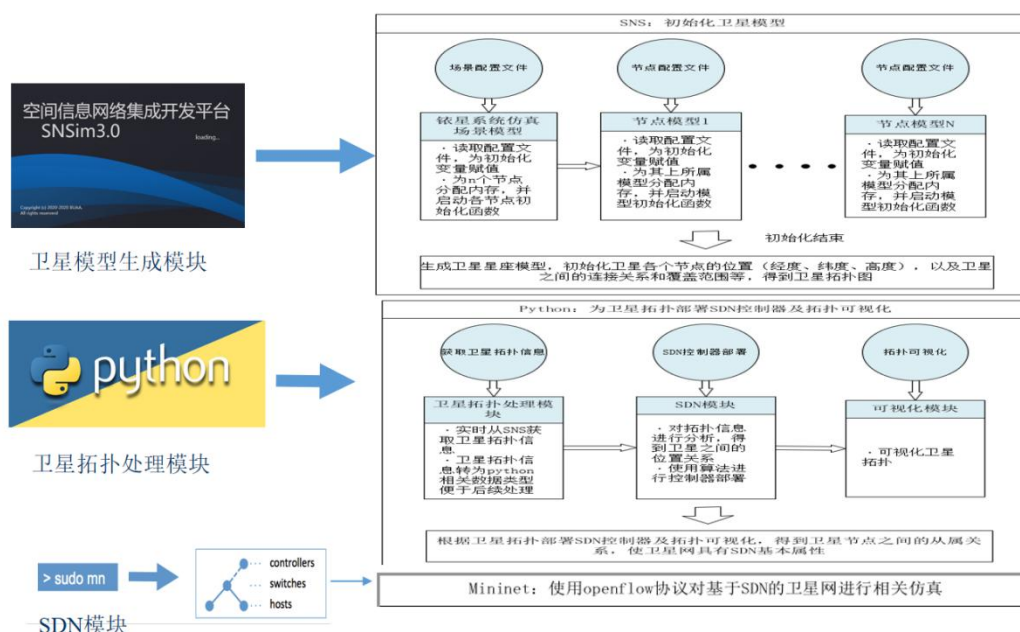


图 15: SDN 卫星仿真流程图

仿真分为三个模块，每个模块由具体的程序负责，卫星生成模块由 SNSim3.0 来根据卫星拓扑的配置文件生成卫星模型，然后将拓扑信息传输给卫星拓扑处理模块，卫星拓扑处理模块将接收到的拓扑信息转化为 python 库的数据格式，然后对这些数据进行处理，包括控制部署算法，控制关系切换以及控制信息更新等，并生成相应的表，最后拓扑处理模块将这些表信息传给 SDN 模块，SDN 模块由 mininet 构成，获得表信息后，可以基于表信息使用 openflow 协议进行相关的仿真。上述过程每隔一分钟执行一次，以实现 SDN 卫星拓扑仿真。

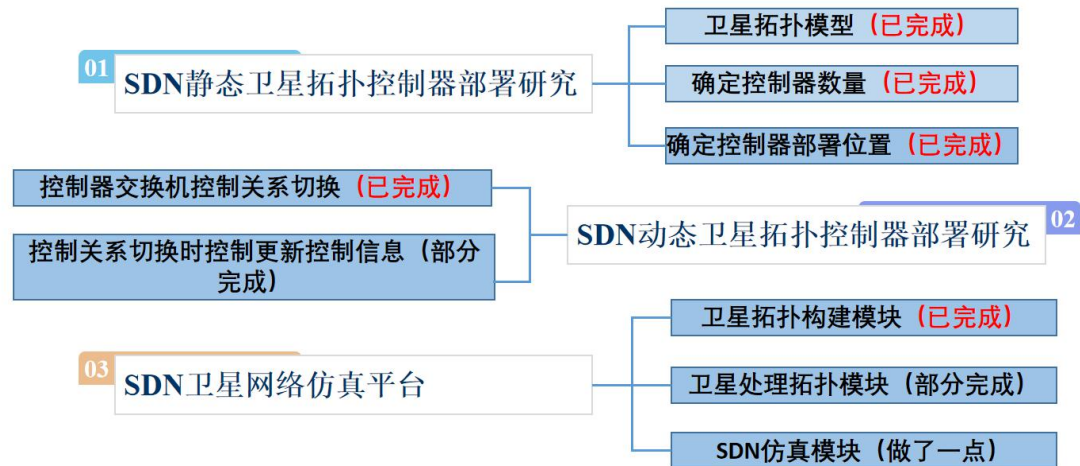
## 第三章 关键技术及难点

本课题的关键技术与难点如下：

- 1、使用时间分片的方式获取网络拓扑与流量信息：使用快照会使用大量星上存储空间，且无法监测节点、链路故障；实时获取的方式又太消耗资源，时间片方式是前两种方法的均衡。
- 3、用户计算控制器与交换机路径的路由算法：需要一种考虑链路时延的路由算法，该算法需要计算出控制器与交换机的最小延迟路径，用于控制器部署决策
- 4、控制器部署决策：根据网络状态，以流响应时间短、为目标，求解网络区域划分与控制器位置选择
- 6、在控制其切换时，控制信息可能没法即是更新而造成丢包，需要衡量切换导致丢包带来代价。

7、仿真软件：mininet 本身不支持动态网络需要对其进行改造。

## 第四章 完成情况



## 第五章 预期目标和进度安排

### 5.1. 进度安排

| 时间                     | 主要研究内容                                   |
|------------------------|--|
| 2021 年 7 月-2021 年 9 月  | 完善控制器之间交互信息方案，将卫星动态拓扑控制器部署方案应用于仿真场景，测试性能 |
| 2021 年 9 月-2021 年 12 月 | 完成整个工程的调试，撰写专利，撰写毕业论文，准备答辩               |

## 第六章 参考文献

[1] B. Heller, R. Sherwood, and N. McKeown, "The controller placement problem," in Proc. ACM 1st Workshop Hot Topics Softw. Defined Netw., 2012, pp. 7–12.16.

- [2] M. Obadia, M. Bouet, J.-L. Rougier, and L. Iannone, “A greedy approach for minimizing SDN control overhead,” in Proc. 1st IEEE Conf. Netw. Softw. (NetSoft), 2015, pp. 1–5.
- [3] Y. Hu, T. Luo, N. C. Beaulieu, and C. Deng, “The energy-aware controller placement problem in software defined networks,” IEEE Commun. Lett., vol. 21, no. 4, pp. 741–744, Apr. 2017.
- [4] 贾梦瑶, 王兴伟, 张爽, et al. 基于软件定义网络的卫星网络容错路由机制[J]. 计算机应用, 2019, 39(6):1772-1779.
- [5] Jia M, Zhu S, Wang L, et al. Routing algorithm with virtual topology toward to huge numbers of leo mobile satellite network based on sdn[J]. Mobile Networks and Applications, 2018, 23(2): 285-300.
- [6] Guo A, Zhao C, Xu F, et al. LEO satellite routing algorithm in software defined space terrestrial integrated network[C]//2017 17th International Symposium on Communications and Information Technologies (ISCIT). IEEE, 2017: 1-6.
- [7] Bao J, Zhao B, Yu W, et al. OpenSAN: A Software-defined Satellite Network Architecture[J]. ACM SIGCOMM Computer Communication Review, 2014.
- [8] Tang Z, Zhao B, Yu W, et al. Software Defined Satellite Networks: Benefits and Challenges[C]ComComAP 2014. 0.
- [9] Hu H, Zhang S, Tang B. LEO SOFTWARE DEFINED NETWORKING BASED ON ONBOARD CONTROLLER[C]//2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP). IEEE, 2018: 1068-1071.
- [10] Papa A, De Cola T, Vizarreta P, et al. Dynamic SDN Controller Placement in a LEO Constellation Satellite Network[C]//2018 IEEE Global Communications Conference (GLOBECOM). IEEE, 2018: 206-212.
- [11] 刘亮, 移动卫星网络的端到端高效可靠传输研究, TN927.23