

北京交通大学

硕士专业学位论文

基于流量预测的5G通信网络优化与系统研究

5G Communication Network Optimization and System Research
Based on Traffic Forecast

作者：毕霄

导师：郭宇春

北京交通大学

2021年5月

学位论文版权使用授权书

本学位论文作者完全了解北京交通大学有关保留、使用学位论文的规定。特授权北京交通大学可以将学位论文的全部或部分内容编入有关数据库进行检索，提供阅览服务，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。同意学校向国家有关部门或机构送交论文的复印件和磁盘。学校可以为存在馆际合作关系的兄弟高校用户提供文献传递服务和交换服务。

（保密的学位论文在解密后适用本授权说明）

学位论文作者签名：

导师签名：

签字日期： 年 月 日

签字日期： 年 月 日

学校代码：10004

密级：公开

北京交通大学

硕士专业学位论文

基于流量预测的 5G 通信网络优化与系统研究

5G Communication Network Optimization and System Research
Based on Traffic Forecast

作者姓名：毕霄

学 号：19125002

导师姓名：郭宇春

职 称：教授

工程硕士专业领域：电子与通信工程

学位级别：硕士

北京交通大学

2021 年 5 月

致谢

本论文的研究工作是在我的导师郭宇春教授的悉心指导下完成的，郭宇春教授开拓的学术思维、科学的工作方法和严谨的治学态度对我产生了极大的影响。当我在科学研究中遇到困难时，郭老师给予了我许多帮助，同时培养了我认真严谨的科研态度以及面对困难积极进取的精神。由衷感谢郭老师这两年来对我在科研上的指导和生活上的关心。

同样感谢陈一帅老师对本次科研工作的帮助和指导。读研期间，多次和陈老师一起讨论学术问题，陈老师总能提出一些新颖独到的见解，帮助我解决心中的疑惑。陈老师饱满的生活热情和积极的工作态度将会一直是我的学习榜样。感谢实验室的所有老师，感谢赵永祥、李纯喜、郑宏云、张立军、孙强等老师在我研究生阶段对我的帮助，在此向各位老师表示衷心的感谢。同时感谢李想师兄、孙欢师姐、戚宇航师兄、王珍珠师姐、于兹灏师兄、冯梦菲师姐、对我学习过程中的无私帮助。同样也感谢张纯、王冰、黎思斯等同学对我学习和生活中的帮助，感谢你们陪我度过这段难忘的学习生涯。感谢一直无微不至地关心、支持我的父母和其他亲人朋友，正是他们不断的鼓励和无私的付出，才使得我不断克服科研中的困难，并顺利地完成学业，成为社会的有用之才。

最后感谢北京交通大学这两年来对我的栽培，感谢电子信息工程学院为我提供的学习机会，感谢网络与安全实验室为我提供了良好的环境去不断试验和创新。

摘要

5G 传输网需要基于现有的 4G 传输网进行演化。在演化过程中, 由于基站升级和人们对网络需求的增长, 传输网中的部分节点接入流量过大, 导致包含该类节点的接入网负载过高, 使整个传输网络出现负载不均衡问题。在这种情况下, 改变现有基站之间的拓扑连接关系, 将高负载网络中的节点让渡给低负载网络, 是一种成本较低、耗时较短的网络负载均衡优化方法。本文为了解决负载不均衡问题, 研究拓扑优化算法来充分探索拓扑解空间, 并且为了保证优化后的拓扑结构在未来的使用中持续有效, 拓扑优化算法评估拓扑不能只基于历史数据, 还要考虑未来流量变化趋势, 所以需要研究流量预测。

本文使用真实电信网络的拓扑数据和流量数据, 设计了一套基于时空预测的传输网拓扑优化系统, 该系统使用历史网络流量和网络拓扑中的空间信息来预测节点的未来流量。系统以预测流量和历史流量作为拓扑优化的依据, 使用遗传算法探索拓扑解空间, 寻找负载均衡的网络拓扑。本文主要贡献如下:

(1) 本文提出在传输网中使用时空预测, 并考虑到目前的时空流量预测模型在衡量具有全局相似、局部相异特征的流量时序相似性时, 缺少有效手段, 导致流量预测不准确。为此, 本文创新性地使用动态时间规整来计算两个流量序列的相似距离。该方法通过自动扭曲时间轴计算最小相似距离, 解决了局部相异所导致的相似距离无法正确反映相似性的问题。实验结果表明: 在传输网中, 相比较于传统非时空预测模型, 本文提出时空预测模型提升了预测准确度 6.7%。

(2) 本文提出了基于随机让渡节点的遗传算法, 解决了传统遗传算法的随机连边所导致的传输网节点连边违反约束条件的问题。本文提出的遗传算法在遵循约束条件的情况下全局搜索解空间。接着, 通过将节点分层分类, 构建等效节点集合, 降低节点间依赖性, 辅助遗传算法。最后, 设计了堆栈式深度优先算法和递归式去冗余算法将整个拓扑划分成数个子网络, 为拓扑优化算法提供基础。这些方法最终在真实大规模网络拓扑上成功实现了网络优化。实验结果表明: 随机让渡节点的遗传算法优化的拓扑比启发式让渡节点算法优化的拓扑, 负载均衡指标提升了 8.4%。

(3) 综合上述两个方法, 本文成功实现了一套基于时空流量预测的真实 5G 传输网拓扑优化系统。实验结果表明, 相比较于原网络拓扑, 该系统优化后的网络拓扑, 在承载历史流量和未来预测流量情况下, 负载均衡指标提高 7.05%, 证明了本论文提出的基于时空预测的网络拓扑优化系统的有效性。

图 26 幅, 表 10 个, 参考文献 41 篇。

关键词: 时空流量预测; 网络拓扑优化; 动态时间规整; 遗传算法

ABSTRACT

5G transmission networks need to evolve based on existing 4G transmission networks. During the evolution process, because of the upgrade of base stations and the growth of people's demand for the network, some nodes that are in transmission networks access too much traffic. This results in a high load on the access network and it also leads to an unbalanced load on the entire network. In this situation, changing the topological connection relationship between existing base stations and transferring nodes in a high-load network to a low-load network is an effective way to improve the network load in a short period of time at a lower cost. For solving the problem of unbalanced load on the entire network, this thesis need to study topology optimization algorithms to fully explore the topology solution space. In order to ensure that the optimized topology will continue to be effective in future, the topology optimization algorithm cannot be based on historical data alone. Future traffic trends also need to be considered. That is why we need to study traffic prediction models.

This thesis uses real telecommunication network topology data and traffic data to design a transmission network topology optimization system. This system based on spatio-temporal prediction, which uses historical network traffic and spatial information in the network topology to predict the future traffic of nodes. It also uses the predicted traffic and historical traffic as the basis for topology optimization, and uses genetic algorithms to explore the topology solution space to find a load-balanced network topology. The main contributions of this thesis are as follows:

(1) This thesis proposes to use spatio-temporal prediction in transmission network, and considers that current spatio-temporal traffic prediction models lack effective method in measuring the time series similarity that has global similarity and local dissimilarity characteristics. Similarity distance that cannot effectively reflect similarity results in inaccurate traffic prediction. To solve this problem, this thesis innovatively uses a dynamic time warping method in the spatio-temporal traffic prediction model to calculate the similarity distance of two time series. Dynamic time warping automatically warps the time axis to calculate the minimum similarity distance and solves the problem, which means similar distance cannot correctly reflect the similarity caused by local dissimilarity. Experimental results show that the method improves the accuracy of traffic prediction by 6.7% over the traditional non-temporal

prediction model.

(2) Firstly, this thesis proposes a new genetic algorithm based on random node transfer to solve the problem, which traditional genetic algorithm violates the connection constraints between nodes due to the random connection of the edges. The new genetic algorithm based on random node transfer can fully search the solution space while the constraints are followed. Secondly, the set of equivalent nodes is constructed by hierarchically classifying nodes to reduce the dependency of inter-node, and it also assists genetic algorithm. Finally, a stack-based depth-first algorithm and a recursive de-redundancy algorithm are designed to divide the whole topology into several sub-networks. Above algorithms provide the basis for the topology optimization algorithm. These methods finally achieve successful network optimization on a real large-scale network topology. The experimental results show that the genetic algorithm based on random node transfer improves the load balancing index by 8.4% compared with the heuristic node transfer algorithm.

(3) By combining the above two methods, a real 5G transmission network topology optimization system based on spatio-temporal traffic prediction is successfully implemented. The experimental results show that the optimized network improves the load balancing index by 7.05% while carrying historical traffic and future predicted traffic. This proves the effectiveness of the system proposed in this thesis.

26 figures, 10 tables, 41 reference articles.

KEYWORDS: Spatio-Temporal Traffic Prediction; Network Topology Optimization; Dynamic Time Warping; Genetic Algorithm

目录

摘要	iii
ABSTRACT.....	iv
1 引言	1
1.1 研究背景和意义	1
1.2 国内外研究现状	3
1.2.1 深度学习的时空预测	4
1.2.2 遗传算法	6
1.3 研究内容及主要贡献	7
1.4 论文组织结构	9
2 技术背景	11
2.1 开发平台	11
2.1.1 编程开发平台	11
2.1.2 数据处理平台	11
2.1.3 深度学习平台	12
2.2 网络流量预测	12
2.2.1 深度学习的流量预测	12
2.2.2 时空流量预测	14
2.3 遗传算法	16
2.3.1 传统遗传算法	16
2.3.2 自适应遗传算法	17
2.4 传输网	18
2.4.1 传输网拓扑结构	19
2.4.2 子网络划分	19
2.5 时间序列相似度衡量	21
2.5.1 欧几里得距离	21
2.5.2 动态时间规整	21
2.6 本章小结	23
3 系统设计与数据分析处理	24
3.1 设计需求	24

3.2	系统设计	24
3.3	数据集介绍及处理	26
3.4	数据分析	27
3.4.1	邻接节点流量的相似性	27
3.4.2	相似数据时间错位问题	29
3.5	本章小结	30
4	基于时空预测的网络拓扑优化	31
4.1	问题与解决思路	31
4.2	时空预测模型	32
4.2.1	动态时间规整相似距离	32
4.2.2	时空预测模型	34
4.2.3	训练流程和参数设置	36
4.2.4	预测模型结果及分析	37
4.3	划分拓扑子网络	40
4.4	随机让渡节点的遗传算法	43
4.4.1	节点分层分类	43
4.4.2	随机让渡节点的遗传算法实现	45
4.4.3	拓扑优化结果与分析	48
4.5	本章小结	52
5	结论	54
5.1	本文工作总结	54
5.2	未来工作展望	55
	参考文献	56

1 引言

1.1 研究背景和意义

近年来,随着人们生活习惯的改变和通信技术的进步,传输网接入的流量具有增长趋势,这种趋势给传输网的负载均衡性带来了挑战。流量增长原因有两方面:一方面,随着 4G 技术成熟,移动互联网人口增长迅速,短视频、网络直播等大流量业务快速发展,更多的移动终端和更大的流量接入到了传输网中,对接入环路容量提出了更高要求;另一方面,5G 传输网是基于 4G 传输网平滑演进的,演进的方法是通过在现有的传输网基础上升级 4G 基站,或者添加新的 5G 基站。但 4G 升级基站和 5G 基站的接入流量远远大于传统 4G 基站的接入流量,该类基站具有超大带宽需求,同样对接入环容量提出了更高要求。上述的流量增长趋势导致现有的传输网难以保证网络的负载均衡。这是由于当前传输网在设计和规划时,未充分预测到流量的未来增长趋势和 5G 基站超大带宽需求,导致现有传输网中各个接入环路的容量利用率呈现整体升高趋势。这种趋势在传输网络上是不平衡、不均匀的,一些接入环路受到流量变化的影响大于其它接入环路受到的影响,导致了各个接入环路的容量利用率出现大小不一致的情况,造成了网络的负载不均衡。在负载不均衡的传输网络中,高负载接入环路影响了传输网的通信稳定性,低负载接入环路无法充分利用网络资源。研究传输网负载均衡问题,有助于保证网络稳定以及充分利用网络资源。

目前,针对接入环路负载不均衡的问题,电信运营商在升级基站以增加接入环路容量的同时,还在进行拓扑优化研究,规划网络拓扑结构。相比较于需要花费大量时间经济成本的网络设备扩容扩建,通过改变网络节点之间的连接关系,将高负载接入环路中的节点让渡给低负载接入环路的拓扑优化方法是一种能够快速、低成本改善网络负载均衡性的方法。拓扑优化方法兼具时效性和经济性的优势,这些优势决定了设计一个拓扑优化系统是具有价值的。

但要注意的是,通过改变连边关系的拓扑优化方法是一种中长期的静态负载均衡方法。为了保证优化后的拓扑结构在未来中长期内能保持有效性,拓扑优化需要预测流量的变化趋势,将预测的流量作为算法优化目标函数的输入,通过控制拓扑连边变量,使得优化后的拓扑结构在流量持续变化中能保持负载均衡性。为此,在设计系统时,需要添加流量预测功能。

本文为了解决传输网络负载不均衡问题,设计并实现了一个基于流量预测的

网络拓扑优化系统。系统通过使用现有拓扑数据和流量数据来优化拓扑结构，实现负载均衡，并且系统的输入数据不依赖特定通信技术背景，系统具有普适性。在未来技术突变导致网络负载不均衡时，系统依然可以持续有效的输出负载均衡的拓扑结构。

为了实现系统，需要研究网络拓扑优化算法和流量预测模型面临的问题。

(1)目前传输网拓扑优化面临的问题是智能水平低下、效率低下和不能有效探索拓扑解空间。目前的传输网络优化主要依赖专家经验，通过人工操作完成，属于被动网络运维。这种网络优化可以在一定程度上消除负载不均衡，保障通信网络平稳运行，却存在智能水平低下、效率低下、不能充分探索拓扑解空间问题。而具有较高智能水平，高效率的拓扑优化算法在在过去的研究中也使用，由于通信网络的拓扑优化是一个 NP-hard 问题^[1-2]，常用的拓扑优化算法为启发式算法，启发式算法可以在不利用梯度信息的情况下充分探索拓扑解空间。现代的拓扑优化算法包括遗传算法，人工蜂群算法，蚁群优化算法等，遗传算法在通信网络拓扑优化中使用最为广泛，该算法的并行搜索可以突破领域搜索限制，实现在整个解空间上的分布式抽样和搜索，从而更有效的探索解空间^[3]。基于这样的特点，本文选择使用遗传算法优化网络拓扑。虽然遗传算法能解决传统传输网优化方法面临的问题，但是遗传算法在传输网络拓扑结构中的使用却面临困难。这是由于传输网拓扑结构的设计具有层次性^[4]，节点之间的连接需要考虑距离限制约束、容量大小约束以及连接后节点不能属于多个接入环路的约束。这些约束条件与传统的遗传算法的探索模式有冲突。传统遗传算法将所有的连边编码，使用编码在拓扑解空间中搜索拓扑结构。算法本质是随机连边，这种随机连边会导致以下问题：接入网中的节点连接到超出距离限制的其它节点、高容量节点连接到低容量节点下游、以及接入节点同时连接到多个接入环路。传输网的约束条件给遗传算法的使用带来了困难。为了实现系统功能，需要找出一种在约束条件下使用遗传算法的方法。

(2)传输网络流量预测面临的问题是没有利用空间信息。为了保证拓扑优化功能的准确性和稳定性，需要精准的流量预测，以保证优化后的拓扑不仅能满足当前的负载均衡需要，还能在未来一定时间内保持负载均衡，从而避免频繁地重新进行拓扑优化。由于传输网节点之间具有流量汇聚和流量传递作用，导致节点之间具有紧密的空间联系，而现有的传输网流量预测却没有考虑到网络流量之间的空间联系性，导致流量预测结果不够精确。为此，本文提出在传输网站使用时空流量预测。考虑到网络流量在时空上存在高度动态性和随机性，分布复杂，导致难以准确预测^[5-6]。针对该问题，近年来的研究者使用深度时空流量预测方法，通过数量众多的神经元组成的网络在空间信息和时间信息中抽取更深层次、更抽象的特征，进而为准确捕捉流量特性提供基础。但是现在的时空预测面临着一个问题，现有的时空预

测方法都有一定程度限制，难以有效地利用空间信息^[6]。过去的工作尝试挖掘节点之间的空间特征，涌现了很多方法，并且取得了不错的效果，其中衡量两个节点的相似距离，将相似距离作为图中的边权重以表示空间信息的方法也被使用^[7]。本文使用该方法，通过衡量网络节点流量之间的相似性来表示空间信息。传统的时序相似度量方法有绝对距离、欧式距离等。但是本文在流量预测实验中，发现两个相似流量没办法用绝对距离和欧式距离来准确衡量。这是因为网络状况的复杂性，相似时序数据之间会出现全局相似、局部相异的特征，从而导致绝对距离、欧式距离等以原始时间轴依次计算时序相似距离的方法失效。针对该问题，需要找出一种方法准确衡量流量相似性的办法，从而有效利用空间信息实现准确的流量预测。

本文致力于时空预测中的空间信息挖掘。针对时序数据之间具有全局相似、局部相异的特征，而局部相异导致难以有效衡量节点之间相似性的问题，更换时序相似度量方法，为时空预测提供有效空间信息。同时本文考虑到遗传算法的随机连边探索模式会打破节点之间连接约束的问题，改变遗传算法的随机探索模式，保证节点连边不会打破约束。最终使用能准确衡量节点相似度的方法捕获空间信息，进行时空预测，将预测结果输入到能遵循约束条件的拓扑优化算法中，实现基于时空预测的网络拓扑优化系统，改善网络的负载均衡。整个系统的设计不依赖于特定通信技术背景，系统具有普适性。随着每一次的通信技术的进步，流量业务性质会周期性突变，本文设计的时空预测的网络拓扑优化系统能持续输出负载均衡的拓扑结构，这种特性使得系统具有一定的经济价值和社会价值。

1.2 国内外研究现状

本文为了在不升级网络设备的前提下改善负载均衡，需要对现有节点之间的连边关系进行重组，进行网络拓扑优化。系统需要一个能充分探索解空间的拓扑优化算法。为此，对拓扑优化算法展开了调研。同时为了保证优化后的拓扑结构在未来能保持负载均衡，需要一个能准确预测的流量预测模型为拓扑优化提供依据。为此，对流量预测模型展开调研。

由于网络拓扑优化是一个 NP-hard 问题，优化中一般使用启发式算法。业界中的传统启发式算法为灵敏度方法，设置有效指标后逐步加连或逐步减边。现代的启发式算法包括遗传算法，人工蜂群算法，蚁群优化算法等。其中遗传算法的全局搜索性，隐并行性使其在拓扑优化问题中使用最为广泛。算法的有效性在通信网拓扑优化、电力网拓扑优化中也得到了证明。本文基于上述原因，使用遗传算法对传输网进行优化，并对遗传算法进行调研。

本文系统设计的目的是保证优化后的接入网拓扑结构在未来流量变化趋势中

保持负载均衡,因此需要预测流量为拓扑优化提供优化依据。业界中传统的流量预测为线性预测模型,例如自回归模型,移动平移模型和自回归合成移动模型,但该类预测模型难以处理流量的非线性、突变性特征。后来出现的机器学习的预测方法,例如支持向量机,可以通过拟合流量曲线后进行预测,从而很好地解决流量的非线性和突变性问题。但机器学习的预测方法难以处理大样本数据。近年来可以处理非线性、突变性以及使用大量数据的深度学习预测在众多领域取得了不俗成就,其中的深度时空预测可以提取空间信息辅助时序预测。本文研究的传输网节点流量之间也具有空间关联性。为此对深度学习的时空预测展开研究。

目前,国内外有大量的深度学习时空流量预测研究,而遗传算法在多年的研究中趋于成熟。下面分别对深度学习时空预测与遗传算法的研究展开介绍,并对各个文献的创新点进行简要说明。

1.2.1 深度学习的时空预测

深度神经网络具备可以从数据中抽取高纬度信息的特点,使其可以处理一些复杂任务,如时空预测。时空预测中的时空数据所分布的高纬度空间具有随机性和复杂性,在进行时空预测时不仅要考虑时间上的周期性和连续性,还要考虑流量之间空间维度上的紧密联系性。目前,国内外研究将深度神经网络运用到时空预测任务中去,并在交通流量预测、天气预测、环境检查等领域证明了深度学习时空预测的有效性。

本文研究真实网络拓扑节点的流量变化趋势。考虑到节点流量之间由于流量汇聚和流量传递作用,具有相互影响的空间关系,并且由于节点的流量和人类生活习惯有关,流量具有时间上的周期性和趋势性,本文关注空间特征挖掘和时间序列预测。

(1)空间特征挖掘

在以往时空预测的研究中,缺少对图中节点层次性划分,但实际上图中节点会有不同的层次和重要程度。考虑到传输网节点具有层系划分,整个传输网分为核心网、汇聚网、接入网,关键节点会有较强的流量汇聚和流量传递作用,识别出关键节点在时空预测中具有意义。ConvLSTM^[8]模型设计了一个以时空为依据的关键地区算子,来识别整个空间中的关键地区,实验表明该模型在具有不同层次、不同重要度结构网络中能有效预测未来流量。

在过去研究中,因为相邻节点之间关联性较强,导致了时空预测过度关注局部联系,但连通网络节点之间不存在互相独立节点,不能忽略全局依赖性。针对以往工作只注重空间局部依赖,忽略全局依赖性的问题,ST-CGA^[9]模型使神经网络

能捕捉全局依赖性。空间动态依赖性一直是时空预测的研究重点,考虑到空间依赖性在大多数场景中是动态变化的,文献[10]提出了一种针对动态空间依赖性以及动态时间平移的预测模型,名为基于时空动态网络(STDN),针对时间序列的长期依赖性和时空流量的空间动态相似性,分别使用带有注意力机制的长短期记忆网络来捕获具有长期依赖性的时间信息和使用设置了门控的卷积神经网络来捕获提取空间动态信息。

以往工作中,孤立的考虑节点之间空间关系或边之间的空间关系,但节点和连边的紧密联系性需要被考虑,文献[11]提出了一种分别用两个神经网络预测节点流量和连边流量的基于时空多任务学习框架(MDL)模型,预测节点的模型命名为NODENET,预测边的模型命名为EDGENET。将两个模型的隐藏状态进行拼接,一同训练。

流量具有周期性、趋势项和非线性特征,而非线性特征时常是由于外部因素引起的,而空间矩阵的限制导致了外部因素很难被神经网络捕获,传输网络在设计时就考虑了外部因素导致的网络故障,说明了外部因素的重要性,而在网络拓扑优化中通过预测系统考虑外界因素是具有意义的,文献[12]为了在时空预测中考虑不包含在输入给神经网络的边权重中的外部因素,提出了一种多分枝预测模型来预测城市短期交通流量的方法,将天气、突发交通事故、等突发事件和节假日和时空信息送入全连接层。为了模拟当前路段和相邻路段之间的空间相关性,将多层全卷积框架用于互相关的计算。

空间信息是多维度的,难以全局的捕获空间信息一直是时空预测所面临的主要问题之一。考虑到图中节点之间的可能会有多种关系,多种关系对应了多维的空间信息。需要通过多种关系衡量节点之间的相似性,文献[7]提出多图融合,将代表不同空间信息的多个系数矩阵送入多个图卷积神经网络来捕捉空间信息。而另一个工作中,使用残差网络、递归神经网络和注意力机制的LA-ResNet^[13]可以对无线网络流量的时空特征建模,使用残差网络捕获空间特征。

(2)时间序列预测

本文研究的是传输网流量预测,传输网络流量节点具有明显的人类活动规律,但是以往的深度时空预测中没有考虑由人类活动带来的流量长期依赖,而长短期记忆网络的遗忘门处理这种长期依赖具有优势。ASTGCN^[14]模型针对流量具有人类活动特征,加入了注意力机制通过学习关键周期特征,充分利用了流量的每周每日等特定日期信息,将这种具有长期依赖性的信息配合空间信息准确预测了流量。

传统的时空预测使用时空序列,将过去的研究迁移到最新的神经网络中是具有价值的,但图卷积网络难以处理序列,文献[15]针对该问题,提出了一种多角度时空预测模型(DMBST-Net),第一次将时空序列视作图像数据,将数据送入卷积

神经网络用于提取空间数据，通过卷积网络加上全连接层的方法提取了空间信息，再将空间信息送入长短期记忆网络预测流量。

文献[16]提出一种用 LSTM 预测流量，遗传算法为 LSTM 提供超参数的方法，利用遗传算法探索参数空间。

大规模网络中的流量预测面临难以实时预测的问题，可以通过时空相关性提供快速预测的方法，如模型 DeepTFP^[17]。结合卷积神经网络、深度信念网络和循环神经网络的方法^[18]不再单独训练一个节点流量，而是考虑从整体预测流量矩阵。网络流量具有突发性，而突发性的非线性特征导致传统方法难以预测，文献[19]使用小波变换将突发性流量变换成近似时序和细节序列，再使用 LSTM 预测，取得了不错效果。流量具有突发性，在突发性流量预测的问题中，使用并行 LSTM 结构^[20]，一个 LSTM 负责主要预处理，另一个 LSTM 负责处理突发流量也取得不错成果。这些时序预测的方法和思想都可以在时空预测中使用。

1.2.2 遗传算法

遗传算法是一种随机在解空间上探索的优化算法，算法模拟自然界物种的种群进化，算法通过变异、交叉、淘汰不断迭代产生更优种群来解决优化问题。遗传算法的主要优势是群体搜索策略，群体搜索策略可以实现在解空间上的并行搜索，分布式的随机采集信息。遗传算法的研究包括编码、交叉、变异、收敛性四个方面，以下展开介绍。

(1) 编码

本文所要优化的拓扑网络由于存在连边的连接限制，无法将连边编码，需要研究编码模式。遗传算法把一个问题编码成算法所能处理的形式，普遍采用二进制编码。但是二进制编码在算法优化过程中会出现缺乏精准度的问题^[21]，选择使用实数编码能解决这个问题，陈辉等人提出了实数编码的方法^[22]，该方法虽然能精准描述解，但是却面临只能运用于连续量的问题，在实际应用中还需要分析问题后设计编码模式。最终，本文选择的编码模式类似于实数编码。

(2) 交叉

本文所要优化的拓扑网络的约束条件一方面给编码带来难度另一方面也限制了拓扑优化的方法，无法随机连边后需要设计一个节点让渡算法优化网络，而节点让渡的功能属于遗传算法中的交叉变异部分，需要展开研究。遗传算法在搜索中，可能会出现可接受解在不可接受解的领域，导致个体还未探索领域就被淘汰，文献[23]提出了一种行为约束程度多维特征表示方法，解决了不可解邻接最优解所导致的不充分探索的问题，该方法实质是添加一个局部搜索过程。因为遗传飘逸^[24]现

象,在搜索中某些个体会出现等位基因丢失的情况,导致某一模式在种群中呈主导地位,从而种群出现缺乏多样性问题,文献[25]提出了种群局部差异淘汰方法,在差异小的个体之间进行局部竞争,为防止种群多样性缺失,为阻止算法陷入局部最优提供了解决方法。交叉作为随机搜索的最主要操作,其过强的随机性在某些情况中是不利的,Caselles 等人提出了有向交叉算子^[26],通过控制交叉结果落点,为随机探索增加了方向性。

(3)变异

优秀个体需要尽可能的将自己的模式传递下去,而劣质个体需要大概率获得突变,文献[27]提出了自适应的遗传变异概率,针对遗传算法在优化过程中因为遗传飘逸等原因过早陷入局部最优的问题,通过使用自适应变异概率以及种群消减解决早熟问题。

(4)收敛

遗传算法作为优化方法,收敛性是需要优先研究的。而遗传算法作为一个启发式算法其全局收敛性一直备受怀疑,Cremers 等人通过研究有限齐次的 Markov 链^[28],使遗传算法的全局搜索的收敛性首次得到证明。而优化问题中,需要证明是否能收敛到全局最优,文献[29]给出了遗传算法以概率 1 收敛到全局最优的方法,即在每次算法迭代中都保留有足够个体距离的个体保优方法。

目前遗传算法使用较多的有以下几种:保持物种多样性遗传算法、多目标优化遗传算法^[30]、自适应遗传算法和混合算法^[31]。其中保持物种多样性遗传算法的目的是保持个体之间的距离,保存种群多样性是为了种群不会因为遗传飘逸收敛到局部最优,体现为解分布的均匀性。在现实中多数的优化问题是多目标优化,各个目标之间有矛盾需要权衡,多目标优化遗传算法用于优化问题中的多目标优化。自适应遗传算法用于解决早熟问题,该算法在多数问题中被广泛使用。混合算法为不同算法和遗传算法混合使用,比如近年出现的增强学习结合遗传算法^[32]。

1.3 研究内容及主要贡献

本文主要研究基于时空预测的网络拓扑优化系统的实现。首先,为了保证系统的有效性,时空预测模块需要精准预测流量,为此需要挖掘空间信息来辅助时间预测。在前文介绍时空预测的研究现状时,简要介绍了多种挖掘空间信息的方法,本文最终选择衡量节点之间相似度的方法,使用节点之间的相似距离作为边权重来表达空间特征。由于网络环境的复杂性,如何准确衡量流量之间的相似性需要研究。本文的研究方法为分析流量数据,并依据分析结果选择合适的相似度衡量算法。接着,在拓扑优化中,由于复杂网络优化问题难以获取梯度信息,本文采用遗传算法

来优化拓扑。由于本文所研究的传输网拓扑中，节点连边具有约束条件，而传统的遗传算法无法满足使用需求，需要针对性的对遗传算法进行改进。

本文的研究难点在于：

(1)本文的目的是通过获取节点之间的空间信息进行时空预测，但是本文所使用的拓扑数据缺乏有效信息。数据中虽然有节点的经纬度，但是物理距离作为相似性度量在实验中效果不佳，并且使用欧式距离衡量两个节点的流量时序相似性的方法在实验中也效果不佳。需要在研究工作中分析数据，并寻找一种能较准确衡量节点流量相似性的方法。

(2)拓扑网络中节点和连边的数目巨大。本文所使用的拓扑数据中节点有 929 个，连边有 1244 条。在拓扑优化时，难以计算和衡量如此巨大的网络。由于本文所研究的传输网络带有层次性，每个节点都有容量大小、类型属性，可以按照网络的层次性把整个拓扑网络划分成不同的子网络。但使用什么算法划分子网络，以及如何针对拓扑特性改进算法，需要研究。

(3)拓扑网络的连接约束条件和节点之间的依赖性导致难以使用传统遗传算法。这是因为随机连边会导致一个节点随机连接到超出距离限制的节点，或者连接到容量小的节点，或者连接到多个接入环路。这种连接还会导致变动的节点及其下游节点一起出现违反约束条件的情况。为此，需要研究一个方法避免遗传算法在随机探索中的破坏约束条件。

本文针对上述难点，进行的工作和贡献点如下：

(1)本文提出在传输网络中使用时空预测，并使用动态时间规整相似距离表示空间特征，提升了预测准确度。在数据分析中发现，存在相邻节点，其流量波形全局高度相似，但是局部相异。局部相异会导致使用欧式距离或者绝对距离计算相似度时，出现一个极大值和一个极小值计算距离，从而计算的相似距离无法正确反映相似度，进而影响时空预测准确度。因此本文使用了动态时间规整，该方法自动扭曲时间轴计算两个时序的最小相似距离，通过该方法提高了预测准确度。相比较非时空预测的基准模型，本文使用的动态时间规整相似距离的 GCN+LSTM+DTW 时空预测模型在预测精度上提升了 6.7%。

(2)本文使用堆栈式深度优先遍历算法、递归式去冗余算法处理了整个拓扑网络，将整个拓扑网络划分成数个子网络。针对所研究的拓扑网络层次，本文首先使用堆栈式深度优先遍历寻找两点间的所有链路，通过交集去重思想，将找到的链路划分成代表汇聚链路的主链路，以及代表接入链路的副链路。划分主副链路后，由于副链路带有主链路节点，这些冗余节点不便于拓扑优化和计算负载均衡指标，需要去除。考虑到副链路之间也带有层次性，次要接入环下挂在主要接入环下，这种特性导致没办法使用交集去重，为此本文设计递归式去冗余算法。该算法搜寻链路

集合中的最短路径，记录路径节点，判断是否冗余，返回处理结果，去除了副链路中的冗余节点，并划分了主次副链路。接着，通过节点集合关系寻找不成环的下挂接入链路。最终使用算法找出了主、副、下挂链路，将同一主链路下的所有链路定义为子网络，进而将整个拓扑网络划分成了数个子网络，为后面的拓扑优化和指标衡量提供了基础。

(3)针对遗传算法无法使用的难点，本文提出将传统遗传算法的探索模式从随机连边改为随机让渡节点。每次让渡节点时都拆除原有连接，将原有连接继承给符合约束条件的节点。这样的做法保证了在随机探索中，节点连接约束不打破。其次，为了简化算法，将所有节点按容量大小和所属子网络分层分类，从而获得等效点集合，等效点集合中的点可以互相代替。使用分层分类等效点集合，每次随机让渡节点都不需要再寻找节点，而是直接从集合中选择等效点。通过这样的方法，本文成功在拓扑网络上运用遗传算法，并最终实现系统功能。优化后的拓扑结构相较于原始拓扑结构，网络的负载均衡指标提高 7.05%，改善了网络负载均衡性。

1.4 论文组织结构

本文的组织结构如下：

第一章为引言。介绍了论文的研究背景和意义，国内外研究现状，论文的研究内容与贡献。

第二章介绍了本实验所需要的开发工具及其使用方法。同时对流量预测的前置知识进行了梳理和介绍，概括性的介绍了时空预测的图卷积网络和长短期记忆网络。接着对遗传算法进行了介绍，介绍了算法的关键知识。接着对本文所研究的传输网、传输网拓扑及子网络定义进行说明。最后阐述了欧式距离相似度衡量和动态时间规整相似度衡量的原理

第三章对本文所设计基于时空预测的网络拓扑优化系统进行了介绍。首先介绍了系统的设计需求。接着介绍系统的整体框架，介绍了每个模块的功能，说明了各个模块对应的系统需求。接着介绍了本文所使用的两个数据集，说明了数据集的大小，字段以及预处理手段。最后进行了数据分析，截取了部分具有代表性的拓扑网络，根据其连接关系，分析流量特征寻找节点之间的空间时间关系。

第四章对本文所设计系统中的预测模型、拓扑优化算法的实现过程展开介绍。首先介绍了如何将两个流量时序建模后进行动态时间规整搜寻两个时序的最小相似距离，并以该相似距离为矩阵元素生成空间特征矩阵。介绍了时空预测模型，说明各个神经网络的作用以及选取该神经网络的理由。展示模型的训练过程和参数设置等工作，设置对照组进行实验，对结果进行了分析。接着介绍了本文如何划分

整个拓扑网络为数个子网络。最后说明本文是如何在具有约束条件的拓扑图上使用遗传算法。将节点分层分类，减弱节点之间的依赖性，然后设计一个交换节点的算法，之后开始设置适应度函数，设置遗传算法的个体距离公式以及自适应变异算子，运行遗传算法。最后对遗传算法的效果进行分析，以及设置对照，验证了本文所设计系统的有效性。

第五章对本文工作进行了总结概括，主要说明了本文工作以及面临的难点和解决方法以及解决方法的效果。列举了本文贡献，说明贡献的意义。最后对工作可改进研究的方向做出了说明。

2 技术背景

本章介绍工作相关的技术背景。介绍了开发平台所使用的语言、环境、开发库以及深度学习框架。对时空预测所使用的神经网络展开说明。介绍了遗传算法的使用流程，以及说明了自适应遗传算法的特点。对本文所研究的传输网进行介绍，对其现有设计模式和拓扑结构进行了简要说明。最后介绍了时间序列相似度衡量，为挖掘空间信息提供支持。

2.1 开发平台

本节主要介绍工作中使用到的编程开发所使用工具，数据处理工具和深度学习框架。

2.1.1 编程开发平台

Python 作为一个高度集成化的面向对象语言，其简单的语句，完整的基础代码库，清晰的表达使 Python 成为了众多数据分析、人工智能科学家的首选开发平台。这为 Python 语言带来了科学研究生态环境。在良好的开发环境下出现了众多扩展库和框架，在机器学习、深度学习、大数据处理等热门领域得到了广泛应用。

Anaconda 作为一个开源的 R 语言和 Python 语言发行包，包含了众多科学研究领域所需框架的依赖包，并且 Anaconda 提供了方便的管理、安装、更新包的功能。同时 Anaconda 还提供了虚拟环境功能，实现了多版本 Python 语言共存功能，提供了优良的兼容性。

本文选择在 Linux 系统中安装了 Python3.6 及其对应 Anaconda。并使用 Pycharm 集成开发环境，Pycharm 提供了优秀项目管理能力以及编程辅助、调试工具。

2.1.2 数据处理平台

Pandas 是 Python 语言的一个扩展数据分析库，提供数值表格和时间序列的运算操作。Pandas 能高效处理数据、检查数据、更新数据、提供逻辑运算与数据分析功能。本文使用 Pandas 对原始的流量数据和拓扑数据进行预处理，以及统计。

Networkx 是用于研究图结构的 Python 库。可以处理有向图无向图，能将图在多种格式之间进行转换，可以查找子图、点集合、边集合以及获取图的邻接矩阵，

并且能以 2D 或 3D 的方式绘图。Networkx 适合在大型图上运行，契合本文数据。

Matplotlib 用于绘制数组的二维图，用于对数据进行分析 and 可视化。Matplotlib 可以绘制带有交互功能和非交互功能的图。支持曲线、折现、柱状图、饼图。

Numpy 普遍用于神经网络的输入张量数据处理，该类数据一般为高阶维度的数组与矩阵，Numpy 针对张量数据的特征设计提供了一系列的数学运算工具。Numpy 的核心功能具有 n 维度的数据结构类 Narray，可以存储和处理多维度数据，常用于深度学习的数据集处理。

2.1.3 深度学习平台

Keras 是一个基于 Python 语言的神经网络开发库，注重于功能的模块化集成与模块之间的脱耦性，注重于扩展性和便捷性。相比较 Tensorflow 和 Pytorch, Keras 更抽象。由于 Keras 的设计理念，AI 开发者都可以较为轻松的搭建深度学习模型。Keras 包含了深度神经网络开发常用工具，如优化器、非线性激活函数、目标损失函数和多种常用的神经网络层。

2.2 网络流量预测

网络流量由于网络的复杂性和流量节点之间的时空相关性，导致流量的突变特征明显，而传统的线性预测模型和机器学习的自身缺陷，导致难以处理非线性、突变型的流量数据。深度学习网络能从数据中提取出数据的高维非线性信息的特征，使得深度学习网络在复杂多变的网络环境中预测流量具有优势，本节介绍深度学习的流量预测。

2.2.1 深度学习的流量预测

深度神经网络发展快速，类型繁多，其中循环卷积网络(Recurrent Neural Network, RNN)和长短期记忆网络(Long-Short Term Memory, LSTM)^[33]在流量预测方面表现良好，涌现了大量相关研究。

传统神经网络在许多领域表现良好。神经网络需要学习的数据经过多层网络加权处理后到达输出层，层与层之间的连接方式一般为全连接，这种学习机制使得隐藏层之间的神经元可以互相联通，但问题是层内的神经元之间不连通。同层无法交互的特性导致了传统神经网络在处理某些问题较为困难。因为数据在现实中的某些特征需要映射到神经网络同层之间的连接，而同一层神经元的相互独立，导致

了无法捕捉到这种联系。

在自然语言处理工作中,进行文本处理时需要使用上文的信息来推测下文,即上下文之间具有联系。这时就需要一种同层神经元可以进行交互的神经网络,需要用到 RNN。

RNN 中的神经元输出不只与当前输入有关,还与之前神经元的输出有关。RNN 内部机制是保存当前的神经元输出,将保存结果传递给后面的神经元。该机制使得隐藏层中的各个神经元不再相互独立,能进行信息交换。

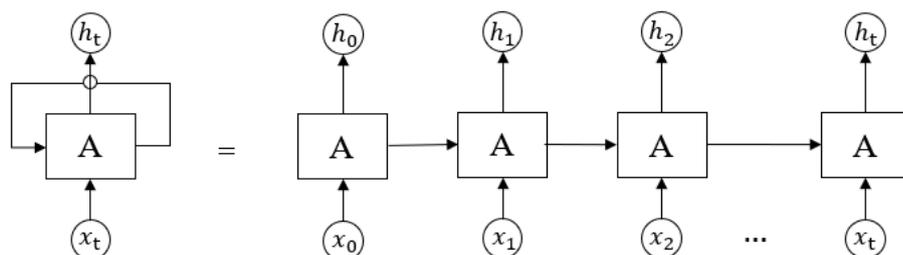


图 2-1 循环神经网络

Figure 2-1 Recurrent Neural Network

图 2-1 为 RNN 的示例图,其中 A 为 RNN 中的神经元, x_i 为隐藏神经层的输入数据, h_i 为隐藏神经层的输出值。当前时刻的神经元使用当前输入数据和之前时刻神经元输出作为神经元的输入。这种信息传递机制决定了 RNN 可以处理具有上下关联的列表和序列。

上述 RNN 可以处理在时间前后具有依赖性数据的问题,但是在处理具有差异性数据时, RNN 的使用会出现问题。例如, RNN 在处理上下文推测工作时,具有较大联系的预测上文和预测下文的间隔较小时, RNN 可以很好的预测,但是当间隔较大时, RNN 却会出现预测结果变差。在理想情况下 RNN 解决的时间依赖长度为无限长,但是在实际中, RNN 不能准确学习到这种具有联系但间隔距离较大的序列信息,导致预测结果不准确。为了解决这个问题,有人提出了 LSTM。

LSTM 设计的目的就是解决时序数据的长期依赖性问题。LSTM 是 RNN 的改进变体,对 RNN 中的神经元内部结构进行了调整。在标准的 RNN 结构中,只有一个简单的模块,该模块使用当前输入和上一个神经元的输出,并将这两个数据整合和输入到当前神经元,再将整合后的结果送入一个激活函数,其过程表示为:

$$h_t = \sigma_h(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \quad (2-1)$$

其中, σ_h 是激活函数, W_{xh} 是将输入数据进行加权整合的权重矩阵, W_{hh} 是邻接的两个隐藏层的神经元之间的连边权重矩阵。LSTM 的封装后的外部信息传递机制与 RNN 相似,但是在神经元内部有不同的处理机制。LSTM 能够解决数据之间的长期依赖性,依靠的是名为细胞状态的信息存储机制。相比较 RNN 的模块 LSTM

添加了三个门结构，分别是遗忘门、输入门和输出门。LSTM 通过使用细胞状态和三个门控机制进行消息传递选择，实现 LSTM 的功能。LSTM 的一个神经元模块如图 2-2 所示。其中， x_t 是当前神经元结构的输入， h_{t-1} 是上一个神经元的输出。

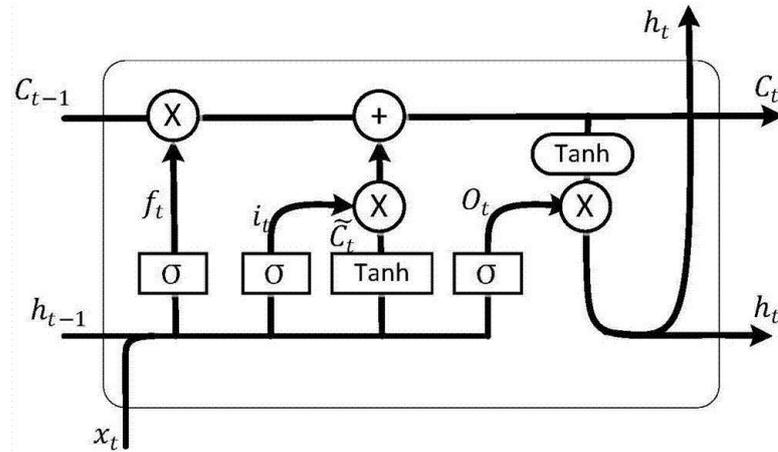


图 2-2 长短期记忆网络内部结构

Figure 2-2 Internal Structure of Long and Short-Term Memory Network

LSTM 通过将上一时刻的神经元输入 h_{t-1} 和当前的数据输入 x_t 进行整合送入到 σ 激活函数中，进而决定遗忘门是否要遗忘之前细胞状态中的信息遗忘门的表达式为：

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (2-2)$$

输入门通过将上一时刻信息和当前信息整合，输入门决定了将多少新的信息加入到细胞状态中，其数学表达式为：

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2-3)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_{iC}) \quad (2-4)$$

输出门决定了当前的神经元结构要输出什么值，其数学表达式为：

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (2-5)$$

$$h_t = o_t * \tanh(C_t) \quad (2-6)$$

LSTM 根据三个门控制信息的遗忘与传递，解决了长期依赖问题。

2.2.2 时空流量预测

当前的流量预测已经不满足单纯提取时间特征来预测流量，开始运用神经网络提取流量时间空间特征，根据时空特征预测数据。而提取空间信息的神经网络为图卷积网络(Graph Convolutional Network, GCN)。

GCN 是基于卷积神经网络(Convolutional Neural Networks, CNN)^[34]的, 相比较 CNN 处理具有时间平移不变性的欧式空间数据, GCN 可以处理不具有时间平移不变性的非欧式空间数据, 该特性使 GCN 在与图结构有关的大量现实问题中获得了广泛应用。与 CNN 提取图片特征功能类似, GCN 的主要功能为提取图结构数据的空间信息^[35], 将图中目标节点的周围节点信息进行聚合。GCN 设计了一种从图数据中提取特征值的方法, 完成了类似 CNN 卷积核的功能。GCN 使用了图谱理论在离散拓扑结构中实现卷积操作, 通过使用图的拉普拉斯矩阵的特征向量和特征值在非欧空间的图结构上将目标节点和邻接节点的信息进行整合。而图上的频谱卷积定义为傅里叶域中由 $\theta \in R^N$ 作为参数的滤波器 $g_\theta = \text{diag}(\theta)$ 与图中节点的标量 $x \in R^N$ 与的乘法, 图谱卷积的定义为:

$$g_\theta * x = U_{g_\theta} U^T x \quad (2-7)$$

其中, U 为拉普拉斯特征向量矩阵 $L = I_N - D^{-\frac{1}{2}} A D^{-\frac{1}{2}} = U \Lambda U^T$, L 为拉普拉斯矩阵, A 是图中的邻接矩阵, 使用增加自环的 A 计算拉普拉斯矩阵, 并对拉普拉斯矩阵进行了归一化。 $U^T x$ 和图的特征值对角矩阵 Λ 共同构成了 x 的图傅里叶变换。使用拉普拉斯矩阵与节点特征矩阵进行矩阵乘能汇聚周围节点信息。但是上式的计算量巨大, 在 Hammond^[36]等人的工作中, 为了减少图谱卷积的计算量, 提出了 $g_\theta(\Lambda)$ 可以通过切比雪夫多项式 $T_k(x)$ 的 k 阶阶段展开近似表达:

$$g_{\theta'}(\Lambda) \approx \sum_{k=0}^K \theta'_k T_k(\tilde{\Lambda}) \quad (2-8)$$

其中, $\theta' \in R^K$ 是切比雪夫系数向量, $\tilde{\Lambda} = (2/\lambda_{max})\Lambda - I_N$, λ_{max} 是拉普拉斯矩阵的特征最大值。切比雪夫递归多项式的定义为 $T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x)$, 其中 $T_0(x) = 1$ 、 $T_1(x) = x$ 。

根据公式(2-7)和公式(2-8), 可以得到:

$$g_{\theta'} * x \approx \sum_{k=0}^K \theta'_k T_k(\tilde{L}) x \quad (2-9)$$

其中, $\tilde{L} = (2/\lambda_{max})L - I_N$ 。可知 $(U \Lambda U^T)^k = U \Lambda^k U^T$ 。但是要注意到公式(2-9)是 L 的 K 阶多项式, 即它仅仅能汇聚中心节点 K 跳的节点信息。Defferrard^[37]等人的工作中定义了图上的卷积神经网络, 通过使用 K 阶部化卷积。当拉普拉斯的阶数为 1 时, 即变为线性函数。当公式(2-9)中的 $\lambda_{max} \approx 2$, 该式简化为:

$$g_{\theta'} * x \approx \theta'_0 x + \theta'_1 (L - I_N) x = \theta'_0 x - \theta'_1 D^{-\frac{1}{2}} A D^{-\frac{1}{2}} x \quad (2-10)$$

其中, θ'_0 和 θ'_1 为过滤器参数, 该类参数在图上共享。通过叠加使用该过滤器可以在图上连续的卷积提取中心节点附近 k 跳节点信息。但在工程实践中, 针对过拟合和

减少矩阵乘法是可以继续限制参数的：

$$g_{\theta'} * x \approx \theta \left(I_N + D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \right) x \quad (2-11)$$

其中 $\theta = \theta'_0 = -\theta'_1$ 。并且由于 $I_N + D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$ 的范围在 $[0, 2]$ 内，所以在深度神经网络中使用这个过滤器时，会导致梯度消失或梯度爆炸。为此需要将 $I_N + D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$ 变为 $\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}$ ， $\tilde{D}_{ii} = \sum_j A_{ij}$ 。

最终使用过滤器进行图卷积进行特征提取可以表示为：

$$H^{(l+1)} = \sigma \left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)} \right) \quad (2-12)$$

其中 \tilde{A} 为增加了自环的邻接矩阵， \tilde{D} 为邻接矩阵的度矩阵， $H^{(l)}$ 为 l 层的图卷积输出，当 $l = 1$ 时 $H^{(1)}$ 为节点特征矩阵， $W^{(l)}$ 为 l 层的权重矩阵， σ 为激活函数。

2.3 遗传算法

遗传算法是一种优化算法，该算法模拟有机生物的遗传过程，通过不断迭代种群以及适者生存的自然选取，获得优秀个体，从而获得优化问题的解。遗传算法使用种群来并行探索解空间，种群中的个体被视为解空间中的一个解。算法根据适应度函数输出的指标衡量每个个体的适应程度，通过设置交叉的选择机率，使得高适应度的个体具有更大机率交叉，传递高适应度个体的模式，使得种群向局部最优解或全局最优解前进。遗传算法通过这样的方式，在不使用梯度的情况下使优化问题的解趋于可接受解。

2.3.1 传统遗传算法

(1) 适应度函数

遗传算法首先需要设置适应度函数，适应度函数是算法中最关键的部分^[38]。适应度函数决定了在淘汰个体时是否能让整个种群的适应度平均值向全局或局部最优解个体的适应度值前进，即适应度函数的设置好坏直接决定了算法的能否收敛。适应度函数接受个体作为输入，返回一个代表个体价值的数值，合理的适应度函数应能真实反映个体的价值。适应度函数应是平滑且具有规律的，不应有太多的局部最大值，局部最大值的个数影响了个体陷入局部最优解的可能性。极端情况为只有一个最大值，最大值为最优个体的适应度，该情况下，算法迭代足够多次后，种群的最优个体将会以概率 1 收敛到全局最优解。

(2) 编码

在遗传算法运行之前，必须为所优化问题的解设计一个适合的编码。搜索空间的大小和问题编码中使用的维度有关。合理的编码能减少搜索空间大小，提升搜索速度。对于长度为 L 的二进制编码，搜索空间大小为 L^2 ，算法对大小为 L^2 的搜索空间进行采样。需要遗传算法的问题，其编码不宜太短，一般少于 30 位编码的问题可以直接使用枚举方法，无需随机程度高的遗传算法，因为遗传算法可能会有冗余探索，不适合在小搜索空间中随机采样。

(3)交叉和变异

遗传算法通常使用交叉和变异机制。交叉需要选择个体对，具有适应度高的个体需要有较大的概率进行交叉，从而将优秀个体的模式进行传递。交叉通常不适用于所有被选中的个体对，需要进行随机选取，随机选取的概率一般选择在 0.6 到 1.0 之间，取决于个体的适应度。对于被交叉操作选择的两个个体，需要将个体的模式进行混合以提供随机探索性，混合方法为编码的交叉，交叉通常使用单点交叉方法。在交叉后进行编码的随机突变。传统的观点将交叉视为变异和交叉中更重要的随机搜索手段。变异仅仅提供了微弱的随机搜索。

(4)收敛性

遗传算法作为优化算法，需要考虑收敛问题。在遗传算法中，这种收敛性体现在算法的迭代运行中，种群的平均适应度是朝向局部最优或全局最优个体的适应度前进的。有许多手段来保证算法的收敛性，如局部竞争、种群密度、自适应变异等。随着种群的收敛，平均适应度将接近最优个体的适应度。

(5)优缺点

遗传算法在迭代运行中，不需要使用梯度信息，而是通过随机探索后直接根据适应度函数来评价种群，对于不可微分但是在解空间中有足够稠密的可接受解的一些 NP-hard 问题，这种不需要梯度，且在一定程度上能保证优化到可接受解的启发式算法具有价值。但遗传算法搜索的可接受解不能是稀疏的，这是因为遗传算法随机搜索方法在巨大的解空间中搜索稀疏解时会有大量的冗余探索。

遗传算法的强大之处在于该技术具有良好鲁棒性和通用性，能成功处理各种问题。遗传算法不能保证找到问题的全局最优解，但通常能找到一个可以接受的解。如果用专门的技术来解决特定问题，那么它们在寻找最终结果的速度和准确性上可能会超过遗传算法。

2.3.2 自适应遗传算法

相比较传统的遗传算法，自适应遗传算法(Adaptive Genetic Algorithm, AGA)对算法的变异设置了自适应算子，每个个体的变异的概率与个体的适应度高低有关。

AGA 通过这种手段保证了算法的收敛性和收敛速度。

AGA 修改了遗传算法中的变异操作，将原来所有个体拥有相同的变异概率的随机探索机制改为了变异的概率与适应度有关的随机探索机制。因为传统的遗传算法会导致优良个体的不公平变异问题，对于优秀个体需要小概率变异，从而能传递其高适应度模式，而对于劣质个体需要增大变异概率，使其能尽可能的改变劣质情况。所以需要设置一个使变异率与个体的适应程度成正比的算子，保证优胜劣汰：

$$P_m = \begin{cases} \alpha(0.5 - f_d^2), & 0 \leq f_d \leq 0.5 \\ \alpha(1 - f_d)^2, & 0.5 \leq f_d \leq 1 \end{cases} \quad (2-13)$$

其中 f_d 为归一化的适应度值， α 为变异率比例系数。

AGA 为了保证对解空间的有效探索，添加了种群密度机制，该机制计算个体之间的距离，通过设置阈值将距离近个体中适应度低的个体淘汰，使得种群中的个体不断变化的同时，使个体均匀分散在解空间中。充分利用了遗传算法的并行探索来探索可能含有最优解的空间。计算两个个体的距离的方法不固定，一般选取欧氏距离。

2.4 传输网

传输网具有层次划分，其层次如图 2-3 所示，主要分为核心层、汇聚层以及接入层。其中核心层主要用于骨干网之间的传输。因为核心层是所有流量的承接者和汇聚者，所以核心设计对有效性的要求严格，设计时需要考虑冗余度，从而保证其有效性。而汇聚层设计目的是承接一部分核心层功能。传输网通过设置汇聚层从而可以提前进行一定程度的流量汇聚，将整个汇聚问题分而治之，减轻了核心层的工作负荷。接入层面向用户业务，接入用户流量。

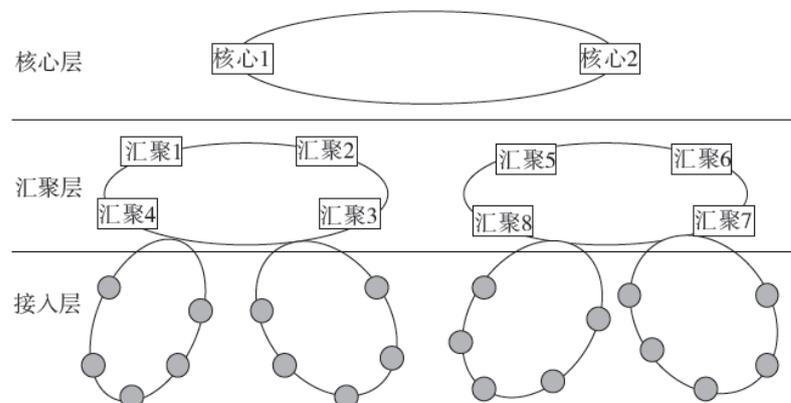


图 2-3 传输网结构

Figure 2-3 Transmission Network Structure

目前,国内三大电信运用商的传输网络拓扑结构建设遵循组网原则^[5]。核心层节点一般控制在 2 个节点。为了保持汇聚层结构的稳定性,不建议添加新的汇聚环。而每个汇聚层节点数量一般控制在 4 至 6 个节点之间。当遇到新增汇聚节点的情况,对现有的汇聚环进行节点补充。汇聚环要求为双归属环,以保证汇聚层具有抗故障能力。而对于接入环,其主环路上的节点数量需要设置限制,数目不易超过 6 至 8 个节点,而每个副环路上的节点不易超过 15 个节点。为了保证接入环的有效性,建议接入环使用双归属环为主。

2.4.1 传输网拓扑结构

为了在经济成本限制下,保证业务的有效性,传输网中的汇聚层网络的拓扑结构设计为双归属环网^[39-40],对接入层网络的拓扑结构设计也建议使用双归属环网。环网是一种网络结构,网络中的设备互相连接成闭环。传输网要求核心层以及汇聚层为环网结构,接入层一般使用环结构,但考虑到经济原因,可以是链状结构。当接入环与汇聚环相连时,如果只有一个节点挂在汇聚节点下,接入环是单归环网,如果有两个节点挂在汇聚节点下,接入环为双归环网。

双归属以及环网的设计目的都是为网络提供冗余度,从而保护网络业务。环结构中的冗余连接为环中节点提供了双向通信的可能性,当环网中的一个节点发生故障,其余节点依然可以通过环网另一个方向的路径保证有效性。环网保证了单个节点故障时的其余节点通信的有效性。而双归属保证了传输网中某一层的一个环网与不同层的环网连接点失效时,依然可以通过另一个连接节点与不同层进行通信,冗余连接为传输网不同层之间的通信提供有效性保障。

在传输网设计和规划中,具有三个约束条件:考虑到经济原因,各个节点之间的连接距离具有限制;节点之间的流量汇聚和流量传递作用也决定了小容量节点只能连接在高容量节点的下游;接入层的拓扑结构一般为环网或链状网络,并且各个接入环相互独立,不存在公共节点,这决定了一个接入节点是不能同时属于两个接入环路的。

2.4.2 子网络划分

在传输网中,汇聚链路和挂在汇聚链路下的主接入环路、副接入环路一同构成了一个接入子网络。本文研究的负载均衡拓扑优化算法所面对的对象就是各个接入子网络。为此需要明确接入子网络的定义,为后文的划分算法提供依据。

本文的传输网拓扑数据中的节点具有一个类型属性。其值为 G,H,J,三者分别

代表了核心层节点、汇聚层节点和接入层节点。结合传输网节点属性以及环形网络的特点，可以将整个拓扑网络中不同节点组成的链路进行区分，区分为主链路、副链路和下挂链路，其中主链路代表了汇聚链路，副链路代表了主接入环路和副接入环路，而下挂链路代表了不成环的接入链路。可以依据区分的链路将整个拓扑网络划分成数个子网络。下面以图 2-4 为例，展开介绍主链路、副链路、下挂链路和子网络的定义。

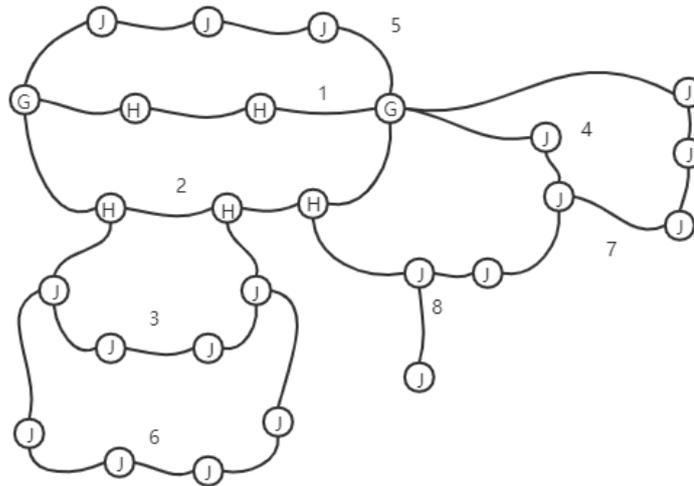


图 2-4 子网络划分
Figure 2-4 Subnetting Dividing

主链路定义分为两种情况。第一种首尾节点都是核心 G 节点，中间节点都是汇聚 H 节点(如链路 1 和 2a)；第二种情况是首尾节点是核心 G 节点或汇聚 H 节点，但中间节点都是接入 J 节点(如链路 3、4、5)。两种定义都要有一个限制条件：首尾节点和中间节点的节点容量值要相同。

副链路首先需要副链路的首尾节点在主链路上，然后定义分为两种情况，第一种情况是首尾节点都是接入 J 节点，中间节点都是接入 J 节点(如链路 6)；第二种情况是首尾一个接入 J 节点，另一个节点为核心 G 节点或者汇聚 H 节点，而中间节点都是接入 J 节点(如链路 7)。两种定义同样有一个限制：中间节点的容量值小于等于首尾节点的容量值。

下挂链路的定义为：挂在某一个 G,H 或 J 节点下的链路，下挂链路中的节点容量要小于等于被下挂节点的容量。

最终可知，子网络定义为主链路加上所有的副链路加上所有的下挂链路。子网络的定义描述了一个接入子网络由汇聚链路和挂在汇聚链路下的主接入环路、副接入环路以及接入链路一同组成。

2.5 时间序列相似度衡量

时间序列本质上是高维数据，高维数据的特征挖掘涉及一系列的挑战。通信网络节点之间的具有流量汇聚作用和流量传递作用，导致了相邻节点之间的流量波形具有相似性。但是由于网络的复杂性，会出现全局相似局部相异的流量时序对，导致时间对应的相似度衡量方法难以有效反映时序之间的相似性。本节详细介绍了欧几里得距离相似度和动态时间规整(Dynamic Time Warping, DTW)。

2.5.1 欧几里得距离

欧几里得距离符合直觉且在多数场景中具有合理性。在处理时序数据时，欧几里得距离是一种常用的相似性衡量方法，该方法通过将两个长度相同的序列之间的点对距离相加后来衡量序列之间的相似度。对于两个长度相同的向量 $X = (x_1, x_2, \dots, x_n)$ 和 $Y = (y_1, y_2, \dots, y_n)$ ，它们的欧几里得距离计算公式为：

$$d(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2-14)$$

描述两个序列之间的相似性，时间对应的欧几里得距离在多数场景中是有效的方法，但是对于时间序列之间出现全局相似局部相异的情况，计算得到的欧几里得相似距离会比实际的最小距离相差很多，不能很好的反映时序之间的相似性。本文所要研究的流量来源于真实的传输网络，其网络复杂性导致了欧几里得距离衡量的相似距离表现较差，最终影响了预测结果。

2.5.2 动态时间规整

动态时间规整用于将两个总体相似，但局部相异的时序数据在时间上对齐，并计算对齐后的相似距离^[41]。

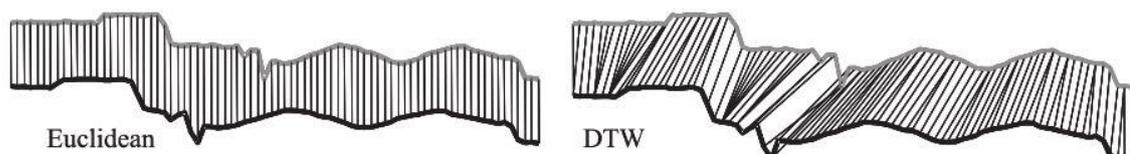


图 2-5 动态时间规整
Figure 2-5 Dynamic Time Warping

图 2-5 对比了使用欧几里得距离计算两个时序相似距离和使用 DTW 改变时序

之间的时间对应关系来计算的最小相似距离。可以看到 DTW 中的一个时刻序列某时刻的点可以对应到另一个时间序列的非同一时刻的点，DTW 通过该方式计算出的距离可以很好的衡量时间错位时序之间的相似度，解决了时序之间全局相似局部相异的问题。

动态时间规整是一个优化方法，该方法设计目的是通过自动扭曲时间，使两个时序数据的形态尽可能的对应。对于两个长度分别为 n 和 m 的时间序列，如式(2-15)与式(2-16)所示，将其表示为 Q_n 和 C_m ：

$$Q_n = \{q_1, q_2, \dots, q_n\} \quad (2-15)$$

$$C_m = \{c_1, c_2, \dots, c_m\} \quad (2-16)$$

动态时间规整的目的就是寻找 Q_n 和 C_m 的最佳对应方式， Q_n 的 i 时刻与 C_m 的 j 时刻的对应关系用 $w = (i, j)$ 表示，使用最佳时间对应关系可以求解两个时序的最小相似距离。为了给两个时序中的各个时间点建立对应关系需要设置一个 $n \times m$ 二维矩阵网格，如图 2-6 所示。矩阵元素 (i, j) 表示了 q_i 和 c_j 的距离，这个距离的计算方法不固定，可根据实际选择。同时每一个矩阵元素 (i, j) 代表了 q_i 和 c_j 对应关系的有无。DTW 可以归结为在二维矩阵中从两个序列的起点找到通往两个序列终点的最小距离路径，该路径代表了 Q_n 和 C_m 各个时刻的对应关系和距离。

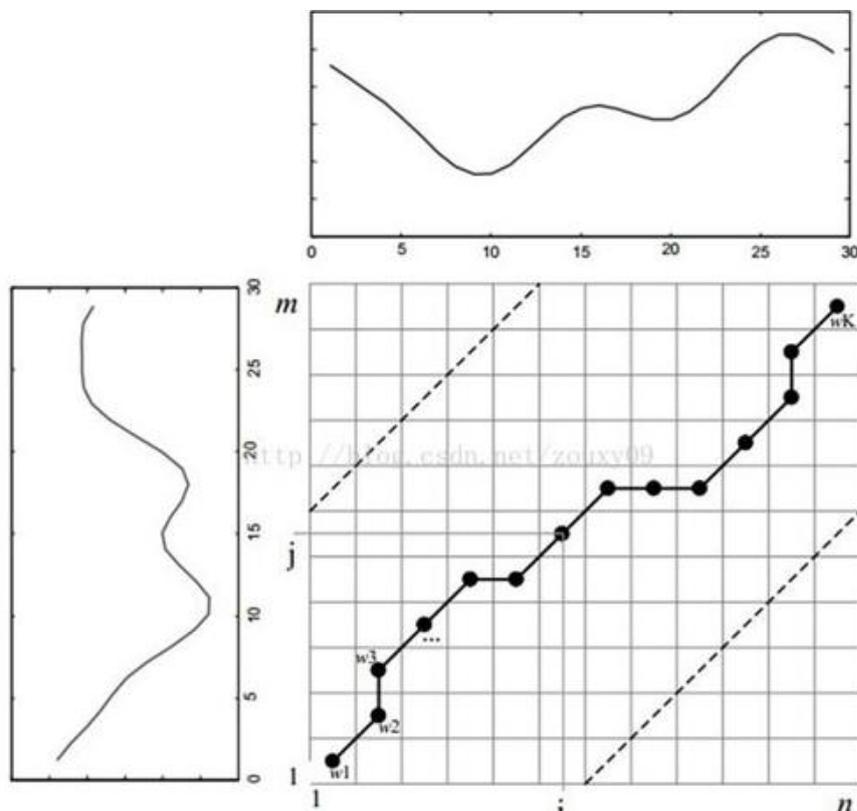


图 2-6 动态时间规整路径
Figure 2-6 Dynamic Time Warping Path

在寻找到路径前，需要给路径的选择设置三个约束，第一个约束是路径的起点必须是矩阵元素(1,1)，这是基于实际考虑的，两个时序是无法对应到零时刻前的数据。其次为了让 C_n 和 Q_n 中的每个节点都有对应关系还需要 w 具有连续性，如果 $w_{k-1} = (a, b)$ 的下一个对应关系是 $w_k = (a', b')$ ，那么为了让 Q_n 和 C_n 中的每一个时刻的数据都有对应点，需要 $(a' - a) \leq 1, (b' - b) \leq 1$ 。最后需要路径满足单调性，如果 $w_{k-1} = (a, b)$ 的下一个对应关系是 $w_{k-1} = (a', b')$ 那么为了保证 W 的点是随着时间前进的而不会出现交叉，需要满足 $0 \leq (a' - a), 0 \leq (b' - b)$ 。

根据连续性条件和单调性条件，对应路径的每一次前进方向被约束。例如路径已经到达了矩阵中的元素格点 (i, j) ，下一个通过的格点只能是以下三种情况之一： $(i, j + 1)$ ， $(i + 1, j)$ 或者 $(i + 1, j + 1)$ 。

满足上述约束的路径数量为指数级，理论上可以选择使用枚举方法寻找路径，但考虑时间复杂度，一般方法为动态规划。使用动态规划选择路径，为了得到最小相似距离，需要定义一个累加距离函数：

$$\gamma(i, j) = d(i, j) + \min\{\gamma(i - 1, j), \gamma(i, j - 1), \gamma(i - 1, j - 1)\} \quad (2-17)$$

该函数表示从矩阵的(0,0)元素开始匹配的时序 Q_n 和 C_m 每到一个点时，之前的所有的点的距离都会累加。最终到达终点 (n, m) 后，这个累加距离 $\gamma(n, m)$ 就是 Q_n 和 C_m 的相似距离。 $\gamma(n, m)$ 代表了 Q_n 和 C_m 的最小相似距离，这种相似距离计算方法可以很好的衡量全局相似局部相异流量时序之间的相似度。

2.6 本章小结

本章主要介绍工作内容相关的技术背景。介绍了本文所使用的开发工具。介绍了时空预测使用的神经网络：图卷积神经网络和长短期记忆网络。对两个神经网络的关键技术卷积核以及细胞状态、三个控制门展开了说明。介绍了遗传算法的设计理念，和关键技术。介绍了本文所研究的传输网以及传输网的拓扑结构。最后针对时间序列相似度衡量方法展开了介绍，介绍了动态时间规整的原理。

3 系统设计与数据分析处理

本章对本文所设计的基于时空预测的网络拓扑优化系统展开介绍。首先，主要分析了系统的设计需求，针对各个需求设计对应系统模块，介绍各个模块的功能及其输入输出。接着，对数据集展开了介绍，介绍了数据集的预处理工作。最后，介绍了流量数据的分析工作。

3.1 设计需求

本文所设计的基于时空预测的网络拓扑优化系统需要满足以下几个方面的需求：

(1)完整的拓扑网络过于复杂，在数据集中拓扑节点与连边数量巨大。考虑到计算以及评估的需求，系统需要将传输网结构的拓扑网络划分为子网络，将子网络作为优化单元以及评测单元。

(2)拓扑网络优化的依据不能只是历史流量数据，本系统的设计目的是为了给未来网络建设提供帮助，因此还要考虑未来流量趋势。为了保证优化后的网络拓扑在未来使用中的有效性，需要准确预测流量，将预测流量作为优化目标的输入。

(3)考虑到拓扑网络节点之间有流量的传递，节点之间会有相互影响，导致节点流量会有一定关联性与相似性。预测流量不能只考虑单个节点的时间信息，还需要考虑空间信息，需要捕捉到节点之间的相似性作为空间信息，进行时空预测。并且由于相似流量之间会出现全局相似局部相异问题，捕获节点相似性时，还需要考虑该问题。

(4)复杂网络的拓扑优化，由于节点和连边的数量巨大，导致传统的拓扑优化方法难以有效探索解空间。针对该特点，需要使用遗传算法，通过变异交叉操作对当前解的领域进行探索，充分探索解空间。

(5)由于接入网节点之间的连接有严格的约束条件，并且拓扑网络中的邻接节点之间具有依赖性，导致遗传算法的随机连边会导致变动节点带动邻接节点一起变动，出现违反约束条件的连边，因此需要设计一种方法减弱节点之间的依赖性。

3.2 系统设计

本节依据 3.1 节提出的设计需求，设计了基于时空预测的网络拓扑优化系统，系统模块对应设计需求，共有五个模块：节点相似性获取模块、流量预测模块、子

网络划分模块、节点脱耦模块和拓扑优化模块。

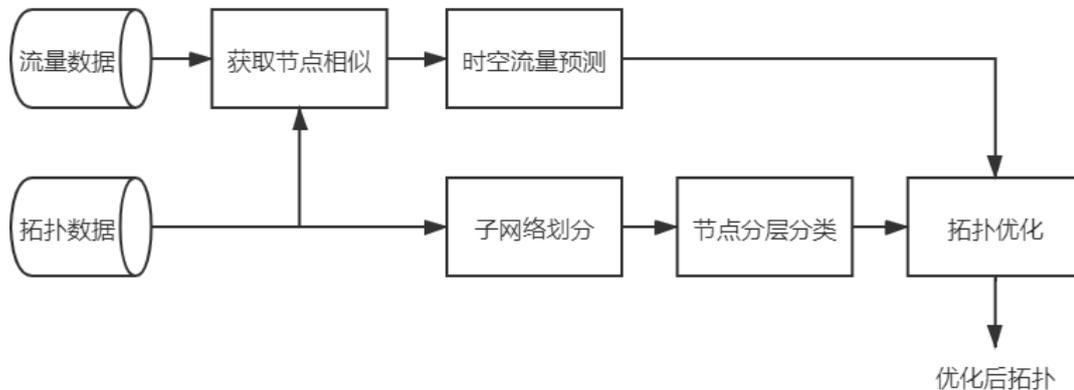


图 3-1 系统框架图
Figure 3-1 System Framework

系统框架如图 3-1 所示，下面对系统的各个模块展开介绍：

(1)获取节点相似性模块的输入为流量数据和拓扑数据，其中流量数据为时序数据，拓扑数据为邻接矩阵。在该模块中计算邻接矩阵所表示的每条连边的两个节点的时序数据的相似性，将计算结果作为边的权重建立新的空间信息矩阵。

(2)时空预测模块的输入为空间信息矩阵和时序数据。系统在该模块中使用空间信息矩阵获取空间信息，使用时序数据获得时间信息，最后联合时空信息预测拓扑网络中每个节点的流量。

(3)子网络划分模块的输入为拓扑数据的邻接矩阵，以及节点属性，属性包括节点编号、容量大小和类型。子网络划分模块根据节点信息以及连边信息将整个拓扑网络划分成数个子公司网络。子网络之间需要有两个汇聚节点保证连通性，但是除了汇聚点外要求子网络之间没有公共节点。

(4)节点分层分类模块的输入为划分好的子网络集合。系统工作流程到达该模块时，每个节点只属于一个子网络，可以将节点所属的子网络看成一个新属性。该模块的功能是利用节点从属于网络和容量大小两个属性对节点进行分类，使同一类的节点互为等效节点，等效点可以替代变动节点维持原来的拓扑结构。系统通过这样的方法降低节点之间的依赖性。

(5)拓扑优化模块的输入为历史流量数据、预测流量数据、分层分类的节点集合，该模块使用拓扑优化算法对拓扑解空间进行探索。将流量数据送入算法迭代中搜寻拓扑结构，使用负载均衡指标对拓扑结构进行评估，最终获得具有良好负载均衡性的拓扑网络结构。

3.3 数据集介绍及处理

本文使用某运营商提供的脱敏流量数据集以及脱敏拓扑数据集，其字段描述如表 3-1 所示。提供的数据包含网络拓扑连接关系数据和节点属性数据。节点属性数据中的流量数据字段记录了每天小时级流量数据，范围从 2020 年 3 月 1 日到 2020 年 3 月 20 日。拓扑数据记录了节点之间的连接关系，且拓扑数据和节点属性数据都含有节点类型、经纬度信息，拓扑节点个数 929，原拓扑连边 1244 条。

表 3-1 流量数据字段描述
Table 3-1 Description of Flow Data

数据集字段	含义
NodeID	节点编号
type	类型，分为：G\HJ
A	容量值
longitude	经度位置坐标
latitude	纬度位置坐标
D	是否为特殊站点
fx_val0	0-1 时节点流量均值
...	...
fx_val23	23-24 时节点流量均值

记录节点之间连接关系的拓扑数据如表 3-2 所示。拓扑数据包含节点对的编号、经纬度、容量大小以及连边的编号。

表 3-2 拓扑数据字段描述
Table 3-2 Description of Topology Data

数据集字段	含义
NodeID_A	节点 A 编号
longitude_A	节点 A 经度
latitude_A	节点 A 纬度
type_A	节点 A 类型
NodeID_B	节点 B 编号
longitude_B	节点 B 经度
latitude_B	节点 B 纬度
type_B	节点 B 类型
NE	连接编号

对节点属性数据集的预处理工作主要是处理缺失值和数据拼接。原数据经过脱敏处理，部分数据的经纬度和流量数据有缺失。本文对于缺失值的处理分为两步：第一步是直接将有缺失的流量数据替换为-1；第二步是将缺失的经纬坐标替换为非缺失经纬的均值。节点属性数据是按照天数进行存储的，存储格式为 csv，当天的 csv 文件包含了拓扑网络所有节点的流量数据。为了将数据打包成张量送入神经网络，首先需要的就是将每天的流量数据进行拼接，然后将拼接好的数据送入内存。本文将处理缺失值和数据拼接工作集成在同一个函数中，该函数使用 Pandas 库读取 csv 文件，对读取的每一个 Pandas 文件按行按列替换缺失值，以节点编号大小进行排序，排序后将多个处理了缺失值的节点数据进行拼接。

对拓扑数据，本文直接使用 Networkx 库，创建图后添加节点列表，将连边的两个节点组合成二元组，送入图中，调用函数获得邻接矩阵。最后将该邻接矩阵用 Pandas 存储为 csv 格式。

3.4 数据分析

本节主要介绍节点之间流量时序相似性分析工作。通过截取整个数据中具有代表性的几个节点，并对这几个节点的流量特性分析，说明流量节点之间的空间关联性和相似流量时序的全局相似局部相异特征。

3.4.1 邻接节点流量的相似性

为了说明节点之间的相似性，并且说明节点之间的空间联系，选取了包括汇聚节点和接入节点的部分拓扑，如图 3-2 所示。

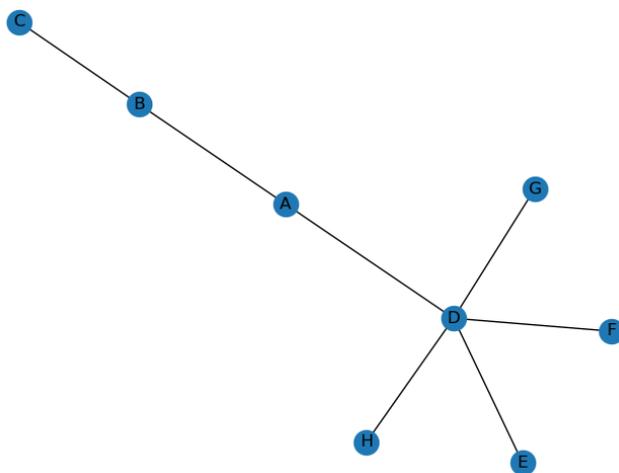


图 3-2 代表性拓扑
Figure 3-2 Representative Topology

选取的代表性拓扑中一共有 8 个节点，其中节点 D 是拓扑中的汇聚节点，其余节点为接入节点。

首先分析节点之间的空间相关性。如图 3-3 所示的是非汇聚节点 A、B、C 一共 20 天的流量波形。

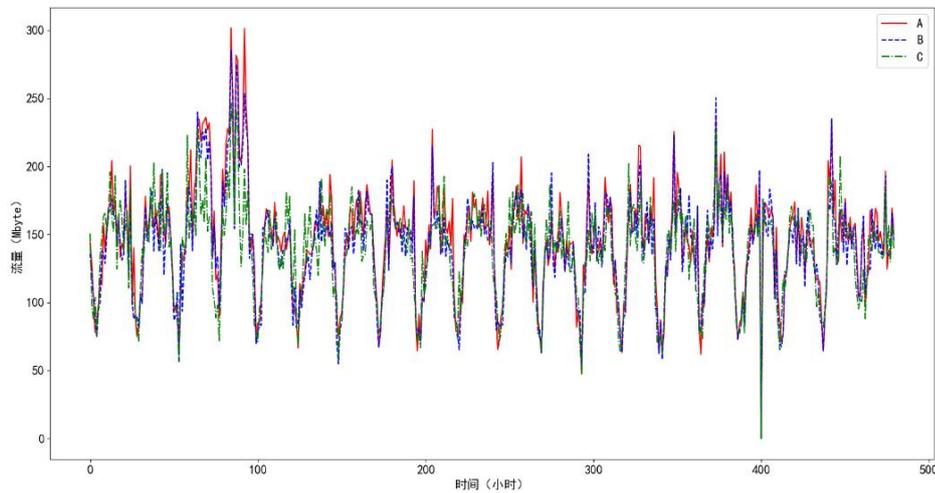


图 3-3 A、B、C 节点流量波形
Figure 3-3 Flow waveform of node A, B, C

可以看到三个节点流量波形具有相似性。三个节点具有以天为周期的周期性和相似的突发性波动，如第 3 天、第 4 天、第 17 天的巨大波动。这说明相邻节点之间是具有类似流量特征的。并且发现节点的流量大小与节点所在位置有一定关联，如图 3-4 所示的第四天流量波形，与汇聚节点 D 跳数越少的节点其流量越大。

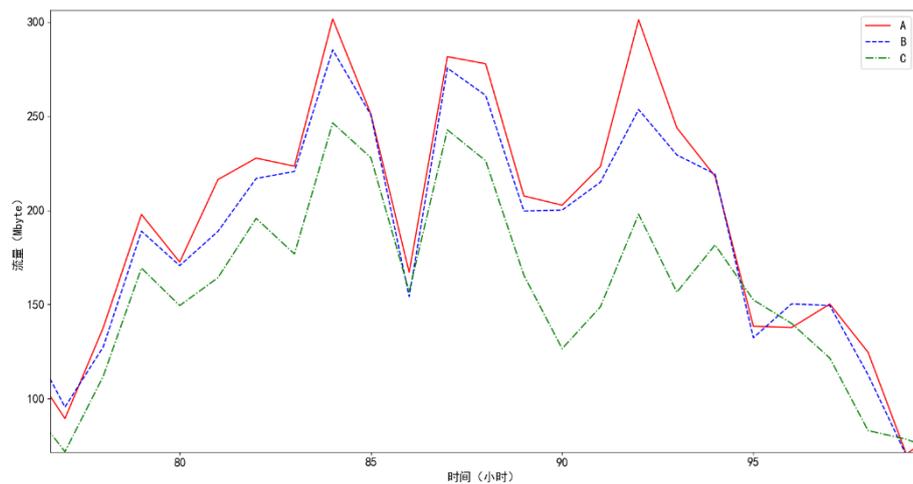


图 3-4 A、B、C 节点第 4 天波形
Figure 3-4 The 4th Day Traffic waveforms of Node A, B, C

这也说明了节点流量之间互相影响，并且这种相互影响表现为流量汇聚和流量传递。接着，分析第 17 天的异常波形，如图 3-5 所示。三个节点出现同时流量

归 0，这可能说明节点之间有一定的地域联系，本文认为这很有可能是三个节点同属某个地区，该地区的外界因素影响到了三个节点。因此本文的时空预测尝试捕获节点之间的物理距离关系。

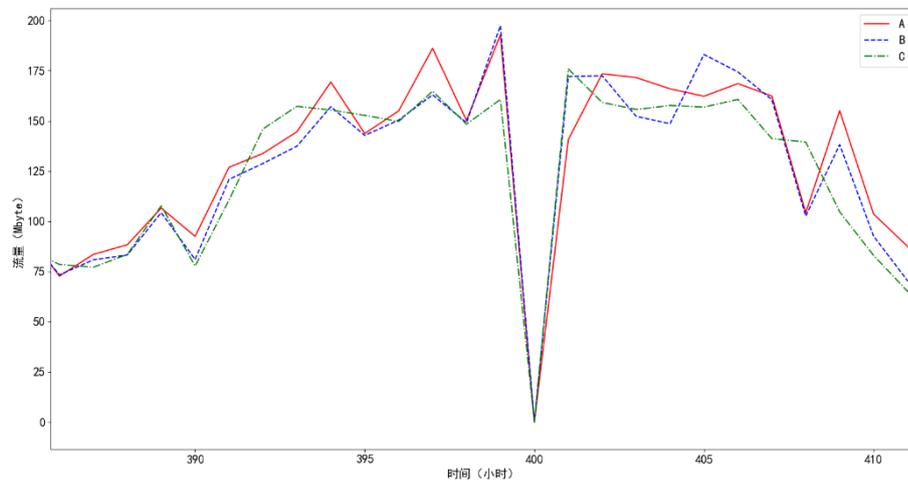


图 3-5 A、B、C 节点第 17 天波形
Figure 3-5 The 17th Day Traffic waveforms of Node A, B, C

3.4.2 相似数据时间错位问题

在数据分析中，本文发现一个问题，如图 3-6 所示。节点流量之间的相似数据点并不是时时对应的，高度相似时序波形之间存在全局相似局部相异现象。汇聚节点 D 的波形与邻接节点的流量波形高度相似，但是节点 D 的波形中会有数个时刻的极大值对应了另一个波形的极小值。

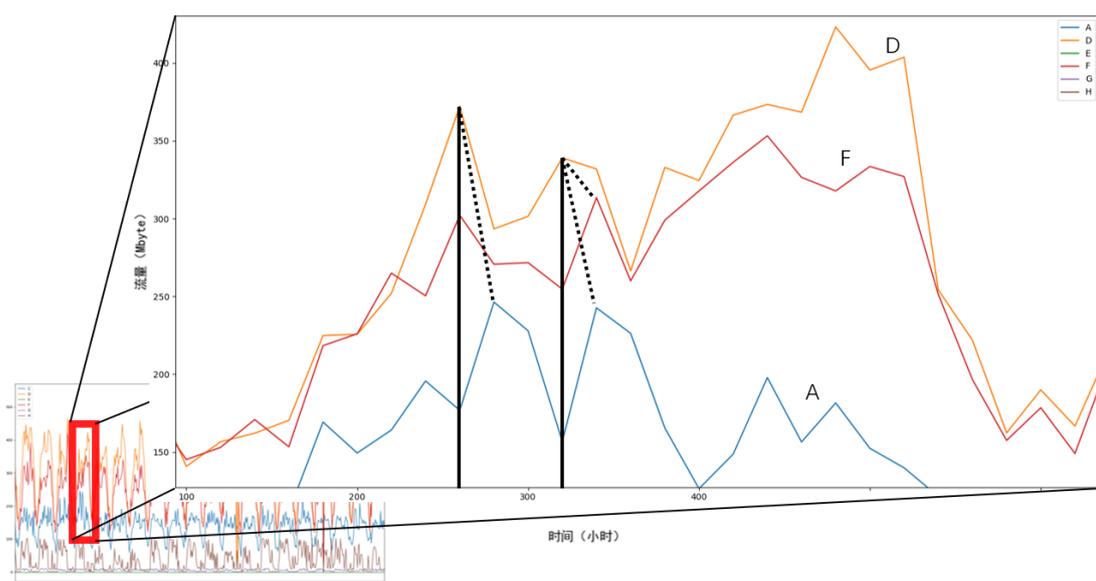


图 3-6 全局相似局部相异流量
Figure 3-6 Globally Similar and Locally Dissimilar Flow

本文发现节点 A、F 和节点 D 高度相似，且其它接入节点也不同程度的和节点 D 相似。可是波形的相似数据却不是时间对应的，图中我们连接了第四天节点 D、节点 F 和节点 A 的数个极大值点，会发现极大值点是不对应的。如果对于这样的数据还继续使用传统相似度计算会出现节点 D 的极大值点对应的是节点 A 的极小值点，这样计算出的相似度是无法准确衡量数据的相似性。

3.5 本章小结

本章主要介绍了基于时空预测的网络拓扑优化系统的设计需求，根据设计需求分析时空预测模型系统实现所面对的难题，找出原因并据此设计模块，规划了模块的输入输出。接着对数据集的组成进行了说明，介绍了数据集的预处理工作。最后截取了部分拓扑，对部分拓扑中节点的流量数据进行了分析。对部分拓扑中相邻节点流量进行了对比，说明了节点流量之间的空间联系，接着对汇聚节点及其下游节点的流量进行分析，说明了流量全局相似局部相异的特点。本章为第 4 章的系统实现提供了技术分析和规划。

4 基于时空预测的网络拓扑优化

本章介绍基于时空预测的网络拓扑优化系统的预测模型和拓扑优化算法实现,介绍了系统实现过程中的关键技术,并对结果进行展示和分析。首先,针对流量时序的全局相似局部相异问题,提出使用动态时间规整来自动扭曲时间,并计算相似距离,将该相似距离作为空间信息送入 GCN+LSTM 的时空预测模型,使用时空预测模型预测了拓扑网络中各个节点的流量。接着,针对拓扑优化和计算容量利用率是以子网络作为单元的要求,将拓扑网络划分成子网络。最后,考虑到拓扑网络节点间存在依赖性,以及在约束条件下难以使用遗传算法的问题,提出了随机让渡节点的遗传算法,将节点分层分类后使得节点在变动时不会影响邻接节点,并且将遗传算法的随机探索模式从随机连边改为了随机让渡节点,实现了在遵循约束条件的情况下使用遗传算法。最后使用预测的流量数据和历史数据,以子网络作为评估单位,使用遗传算法优化了拓扑网络,提高了拓扑网络的负载均衡性。

在介绍本章时,从问题与解决思路、时间预测模型、划分子网络算法和随机让渡节点的遗传算法四部分进行介绍。

4.1 问题与解决思路

拓扑网络中的节点流量如 3.4.1 分析,上游节点对下游节点的流量具有一定的聚合作用。预测流量时不能单独考虑一个节点的流量,需要捕捉该节点周围节点对其的影响,衡量节点之间的关联性。而衡量这种关联性的方法就是计算两个节点流量数据的相似距离,但是以往的做法不能很好的衡量节点流量的相似性。如上文 3.4.2 分析,由于现实中的网络状况复杂,上下游节点的相似数据不一定在时间上对应,会出现相似数据时间错位问题,表现为全局相似局部相异。如果还是按照原始时间轴计算两个时间序列的相似距离,会出现一个极大值与一个极小值计算距离的情况,从而难以有效的衡量两个流量数据之间的距离。最终影响预测精度。

本文使用动态时间规整方法,该方法通过动态规划寻找两个时序数据中每个数据点的最佳对应方式,从而解决相似数据时间错位问题。使用动态时间规整相似度作为两个节点之间的空间信息,将该相似距离取倒、归一化后作为时空预测的拉普拉斯矩阵的矩阵元素,将历史数据送入使用该矩阵的时空预测模型,训练后预测未来流量。

现代离散拓扑优化问题,广泛使用遗传算法。遗传算法具有全局搜索特性,可以并行探索解空间,同时可以对当前解的领域进行随机探索,这些特性使得算法可

以有效探索解空间。传统的遗传算法首先对网络中的连边进行编码，然后通过变异交叉的操作改变编码值，从而达到随机连边的目的，随机连边提供了随机探索性。但是传输网拓扑的连边关系具有约束条件，并且拓扑网络中各个节点是相互关联的，如图 4-1 所示，一个节点的随机连边有可能会产生变动点与该节点直接或间接连接的节点打破约束。具体来说，本课题所要优化的拓扑网络如 2.4 节所述，具有定义和约束条件，而随机连边会导致节点连接到超出距离限制的节点，或者连接到小容量节点，或者同时连接两个子网络。为了在拓扑优化中不打破约束条件，本文将随机探索模式从随机连边改为随机让渡节点，并且将节点分层分类，构建等效点集合来辅助节点让渡。

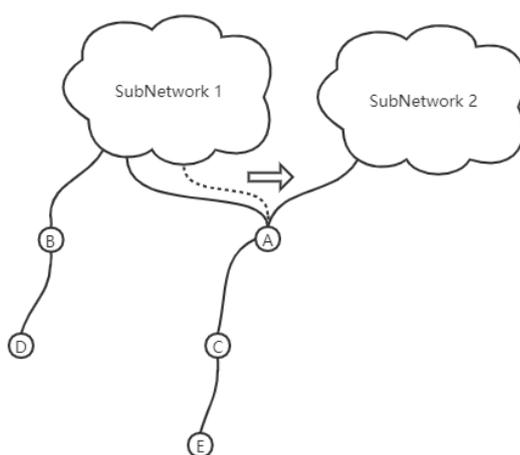


图 4-1 随机连边问题
Figure 4-1 Random connection problem

4.2 时空预测模型

本节介绍了如何使用动态时间规整计算时序相似度，以及如何将相似度转换成空间特征。然后介绍了 GCN+LSTM 模型的使用和训练流程。最后介绍了实验设置、实验结果以及分析。

4.2.1 动态时间规整相似距离

在通信网络中节点流量的输入输出会对周围节点的流量产生影响，而在多变的网络环境中，相似时序会出现全局相似局部相异的特征，该特征会导致非扭曲时间轴的相似度衡量方法无法正确反映时序相似性。本文考虑这样的情况使用动态时间规整计算节点之间流量相似距离，并将动态时间规整相似距离作为 GCN

的空间特征矩阵中的元素。

动态时间规整计算两个长度都为 n 的流量时序相似度时，将一个流量时序数据作为参考流量 Q_n ，另一个作为测试流量 C_n ：

$$Q_n = \{q_1, q_2, \dots, q_n\} \quad (4-1)$$

$$C_n = \{c_1, c_2, \dots, c_n\} \quad (4-2)$$

其中 q_1, q_2, \dots, q_n 和 c_1, c_2, \dots, c_n 代表了时序数据中每小时的流量值大小。

动态时间规整需要确认 C_n 中的每个流量点 c_i 如何对应到 Q_n 中的流量点 q_j ，以及两个流量点之间的距离 $w(i, j)$ ：

$$w(i, j) = \sqrt{(c_i - q_j)^2} \quad (4-3)$$

动态时间规整对应点之间的距离 $w(i, j)$ 有很多计算方法，本文选择的计算方法为两个流量大小的欧式距离。

Q_n 和 C_n 的整体对应关系定义为 W ：

$$W = w_1, w_2, \dots, w_k \quad (4-4)$$

其中 k 代表了两个时序对应数，范围为 $n \leq k < 2n + 1$ 。 W 表示了测试流量中的每个点是如何对应到参考流量的。为了使带有时间错位的相似流量数据尽可能对应， W 的选取应是使相似距离最小的 W_{min} 。

对于本文中两个时序长度相同的情况，流量时序 C_n 和流量时序 Q_n 的可对应数，即 $w(i, j)$ 总个数最少有 n 种，最多有 $2n - 1$ 种。本文通过使用动态规划来求解 W 。为此需要先建立一个 $n \times n$ 矩阵 $A_{i,j}$ ：

$$A_{i,j} = w(i, j) \quad (4-5)$$

矩阵中的 i 行对应 Q_n 的 i 时刻，矩阵中的 j 列对应 C_n 的 j 时刻，矩阵中的每个元素是 $w(i, j)$ 。使用动态规划求解 W 时，首先需要有一个边界条件 $w_1 = (1, 1)$ ，这是因为 C_n 和 Q_n 没办法和未被观测到的流量数据建立对应关系。其次为了让 C_n 和 Q_n 中的每个节点都有对应关系还需要 W 具有连续性。并且为了不让时间对应关系交叉，需要设置单调性。

使用动态规划求解 W 需要定义一个距离累积量 $\gamma(i, j)$ 代表最小相似路径：

$$\gamma(i, j) = d(i, j) + \min\{\gamma(i - 1, j), \gamma(i, j - 1), \gamma(i - 1, j - 1)\} \quad (4-6)$$

其中 $d(i, j)$ 代表了 c_i 和 q_j 的距离， $\gamma(i, j)$ 代表了到达 (i, j) 时的累加距离。每次选择最小累积距离的过程实质上就是在选取 W ，利用式(4-6)，动态规划计算每个子问题的最小距离，最后就得到了两个时序数据的最小相似距离 $\gamma(n, n)$ 。至此获得了两个流

量时序的最小相似距离。需要把该距离设置成代表空间信息的矩阵。

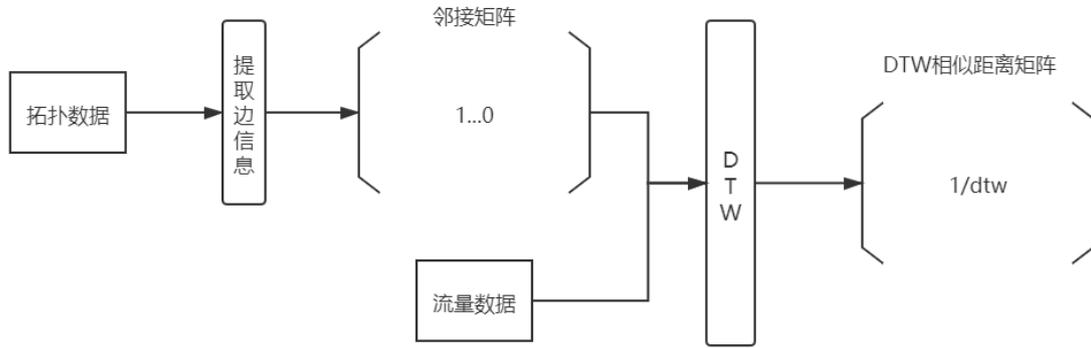


图 4-2 DTW 相似矩阵获取流程

Figure 4-2 DTW Similarity Matrix Acquisition Process

获取整个动态时间规整相似距离矩阵的过程如图 4-2 所示，在生成动态时间规整相似距离矩阵前，首先获得拓扑网络的邻接矩阵，通过邻接矩阵获得所有连边，将连边两端的节点的时序数据作为 Q_n 和 C_n ，送入动态时间规则模块，该模块输出两点流量的动态时间规整相似距离，然后用该相似距离的倒数作为新的拓扑特征矩阵元素：

$$DTW_{i,j} = \begin{cases} \frac{1}{\gamma_{Q,C}}, & A_{i,j} = 1 \\ 0, & A_{i,j} = 0 \end{cases} \quad (4-7)$$

其中 $A_{i,j}$ 代表邻接矩阵第 i 行第 j 列元素， $DTW_{i,j}$ 为动态时间规整相似距离矩阵第 i 行第 j 列元素。由动态时间规整模块输出的相似距离倒数为元素的拓扑特征矩阵，包含了相邻节点流量的相似性，使用图卷积长短期记忆网络进行预测时，该矩阵提供了相邻节点之间的关联性，为时空预测提供了技术基础。

4.2.2 时空预测模型

本小节解释了选取 GCN+LSTM 作为预测模型的理由，然后介绍了模型的输入输出和训练流程。

根据 3.4 节分析，发现节点和邻接节点的流量波形非常相似，具有相同的周期性和相同的波动规律，越靠近汇聚节点的接入节点，其流量越大，而且具有汇聚作用节点的流量波形与邻接的接入节点的流量高度相似。3.4 节的分析说明了相邻节点之间的空间联系紧密，节点之间有流量传递和汇聚作用。因此可以利用拓扑网络节点之间的联系，获取节点之间的空间信息辅助流量预测。GCN 作为能处理非欧式空间数据，能在离散拓扑中捕捉空间信息的神经网络，其对目标节点的周围节点

信息进行聚合的功能符合本文需求。并且 GCN 捕捉空间信息的能力在交通流量预测等工作中得到过证实。因此本文选取 GCN 提取空间特征。

同样在 3.4 节的分析中我们发现，本文中流量数据，具有明显的周期性，且该规律符合人类活动规律，可预测性比较强。在使用 GCN 提取周围节点信息后，需要一个带有预测功能的神经网络，从混合了空间信息的时序数据中提周期信息，LSTM 可以很好的承担该工作。本文使用 GCN+LSTM 的模型框架在时间信息中混入空间信息进行预测。模型框架如图 4-3 所示。

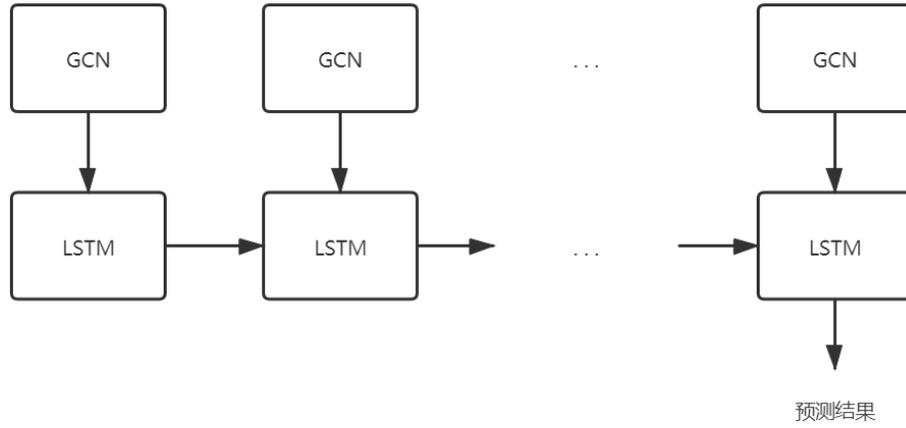


图 4-3 时空预测模型

Figure 4-3 Spatio-Temporal Prediction Model

传统的模型框架中的 GCN 需要两个输入：代表空间信息的邻接矩阵、代表节点特征的节点特征矩阵。本文的空间信息矩阵与邻接矩阵类似，但不同的是连边权重使用上文提取的 DTW 相似距离倒数作为元素。GCN 的第二个输入节点特征矩阵为流量数据矩阵，代表了每个节点的特征，特征的维度就是时序的长度。GCN 通过训练后的网络参数将周围节点的特征聚合到目标节点特征上。聚合方式如下：

$$f_{GCN}(H^{(l)}, DTW) = \sigma \left(\tilde{D}^{-\frac{1}{2}} \overline{DTW} \tilde{D}^{\frac{1}{2}} H^{(l)} W^{(l)} \right) \quad (4-8)$$

其中 $H^{(l)}$ 为 l 层 GCN 的节点特征矩阵， $W^{(l)}$ 为 l 层 GCN 的网络参数矩阵， DTW 为上文 4.2.1 中提取的动态时间规整相似矩阵， \overline{DTW} 为添加了自环的 DTW 矩阵， $\tilde{D}^{-\frac{1}{2}}$ 与 $\tilde{D}^{\frac{1}{2}}$ 为度矩阵用于 \overline{DTW} 的对称归一化， σ 为激活函数，该式用拉普拉斯矩阵 $\tilde{D}^{-\frac{1}{2}} \overline{DTW} \tilde{D}^{\frac{1}{2}}$ 描述空间信息， $H^{(l)}$ 表达节点特征， $W^{(l)}$ 表示聚合时周围节点特征的权重大小。

LSTM 接受 GCN 的输出作为输入，此时的 LSTM 预测过程可以被视为在获知拓扑网络邻接信息 DTW 、节点流量特征 X 的情况下预测下一时刻流量。整个模型的训练可以看作是在学习映射函数 $f_{GCN+LSTM}$ ：

$$[X_{t+1}, X_{t+2}, \dots, X_{t+T}] = f_{GCN+LSTM}(DTW, (X_{t-n}, \dots, X_{t-1}, X_t)) \quad (4-9)$$

4.2.3 训练流程和参数设置

时空预测模型的训练流程如图 4-4 所示。首先本文使用 Keras 深度学习框架搭建网络，然后设置 GCN 和 LSTM 网络的参数，包括激活函数、神经元数量、输入输出维度，接着设置训练参数，包括 `batch_size`、`epochs`、优化器、损失函数。设置完网络后需要处理输入拓扑数据和流量数据，两个数据分别处理成邻接矩阵和流量特征矩阵。将邻接矩阵和流量数据送入 DTW 模块计算相似距离，然后生成 DTW 矩阵。将 DTW 矩阵和流量特征矩阵送入网络进行训练。在每次训练周期内计算损失函数，依据结果反向传播更新网络参数，终获得训练好的模型。

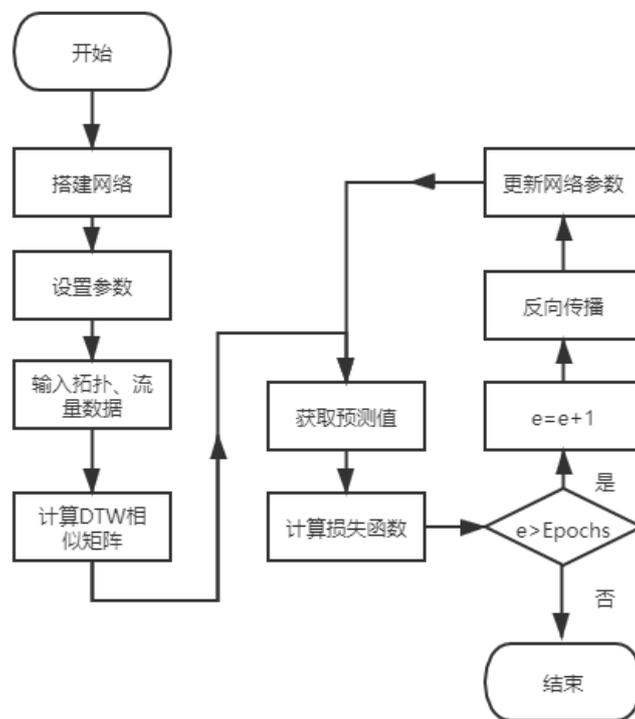


图 4-4 训练流程
Figure 4-4 Training Poccus

本文结合文献以及实验对比，选择的模型参数如表 4-1 所示。确认图卷积层的超参数时，实验对比了隐藏层神经元数、激活函数。最终图卷积神经元数为第一层 60、第二层 40、激活函数为 `relu`。长短期记忆网络的超参数实验对比了隐藏层神经元数、激活函数、训练窗口、预测长度。最终长短期记忆网络层的神经元数设为第一层 200、第二层 100，激活函数为 `tanh`，LSTM 的训练窗口选为 12 小时，每次预测训练窗口的后一小时流量。整个时空预测模型的优化器使用 `adam`，损失函数使用 `MAE`，`batch_size` 大小 60。

表 4-1 时空预测模型参数
Table 4-1 Spatio-Temporal Prediction Model Parameters

参数名称	含义	激活函数	值
learning_rate	学习率	-	0.001
batch_size	批次大小	-	60
optimizer	优化器	-	adam
GCN_1	GCN 神经元数	relu	60
GCN_2	GCN 神经元数	relu	40
LSTM_1	LSTM 神经元数	tanh	200
LSTM_2	LSTM 神经元数	tanh	100

4.2.4 预测模型结果及分析

本文成功搭建 GCN+LSTM 网络，并使用 DTW 相似度矩阵衡量了节点之间的相似距离。预测结果如图 4-5 所示。

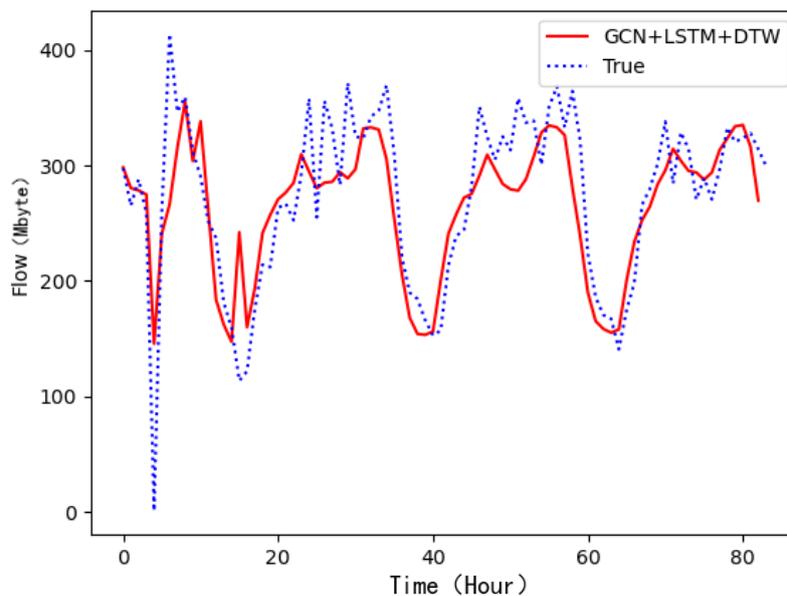


图 4-5 DTW 相似度时空预测结果
Figure 4-5 DTW Similarity Prediction Result

本文基于 3.4 节数据分析，需要研究问题如下：

- (1) 考虑到流量节点之间具有空间关联性，捕捉空间关系是否有助于流量预测。
- (2) 考虑到相邻节点的流量出现了同时跌至为零的情况，是否是因为相邻节点所处区域的外界影响，这是否说明物理距离较近，处于同一区域的节点会有相同的流量模式。物理距离是否是一个有效衡量节点相似性的量。

(3)相似流量会有全局相似局部相异的问题，会出现两个流量数据的极大值与极小值计算距离的情况，导致相似性衡量错误。针对该问题使用 DTW 是否有改善。

为了研究上述问题，本文实现了使用 DTW 相似距离的 GCN+LSTM+DTW 模型，并且设置了四个基准：时间预测模型 LSTM、使用邻接矩阵(Adjacency Matrix, AM)作为拓扑信息的 GCN+LSTM+AM、使用两点间物理距离的倒数作为拓扑信息的 GCN+LSTM+Distance、以及使用流量时序欧氏相似距离(Euclidean Distance, ED)作为拓扑信息的 GCN+LSTM+ED。本文使用平均绝对误差(MAE)和均方误差(RMSE)来对时空预测模型预测精准度进行衡量。平均绝对误差表达如下：

$$MAE = \frac{1}{m} \sum_{i=1}^m |y_i - \hat{y}_i| \quad (4-10)$$

均方误差的公式表达如下：

$$RMSE = \sqrt{\frac{1}{m} \sum_{i=1}^m (y_i - \hat{y}_i)^2} \quad (4-11)$$

本文使用这两个指标衡量所有节点时序数据的预测精准度，然后将所有节点的 MAE、RMSE 进行取平均值，从而衡量整个时空预测模型的性能。各个模型的预测结果如表 4-2 所示。

表 4-2 预测误差
Table 4-2 Prediction Error

预测模型	MAE	RMSE
LSTM	28.742	42.916
GCN+LSTM	25.317	40.419
GCN+LSTM+Distance	26.273	40.832
GCN+LSTM+ ED	26.037	40.999
GCN+LSTM+DTW	25.056	40.019

根据表 4-2 所示结果，可以得到以下四个分析：

(1)将 GCN+LSTM 的结果与 LSTM 的结果相比较，GCN 提取了空间信息。GCN+LSTM 相比较 LSTM, MAE 提升 11.9%, RMSE 提升 5.8%。其中 GCN+LSTM 和 LSTM 的预测值和真实值对比如图 4-6 所示。

GCN+LSTM 相比于 LSTM 对整体预测精度提升较大。GCN 成功的从空间特征中提取了信息，辅助了 LSTM 进行时序预测。证明了在流量网络拓扑中，捕捉空间信息进行时空预测是有效的。

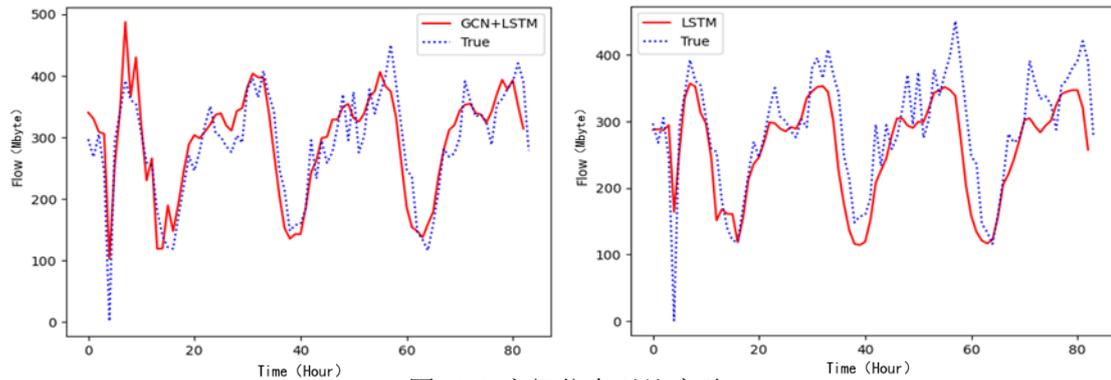


图 4-6 空间信息对比实验

Figure 4-6 Spatial Information Comparison Results

(2)将 GCN+LSTM+Distance 的结果与 GCN+LSTM 进行比较,前者将邻接矩阵中的表示是否有连接的 01 逻辑量改成了已连接的两个节点之间物理距离倒数的归一化值。GCN+LSTM+Distance 的 MAE 降低了 3.7%, RMSE 降低了 1.0%。预测精度有所下降。其中 GCN+LSTM+Distance 和 GCN+LSTM 的预测值和真实值对比如图 4-7 所示。

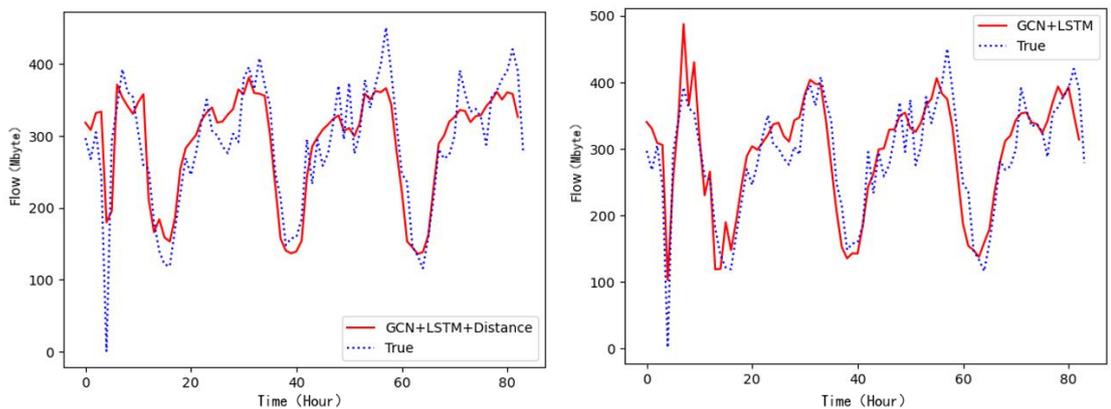


图 4-7 距离相似度对比结果

Figure 4-7 Distance Similarity Comparison Result

该结果说明物理距离在传输网网络拓扑中是没法有效表示空间信息的。物理距离是无法在传输网拓扑中衡量两个流量相似性的。

(3)将 GCN+LSTM+DTW 的结果与 GCN+LSTM、GCN+LSTM+ED 的结果进行比较, MAE 分别提升 1.0%、3.7%, RMSE 分别提升 0.9%、2.3%。预测精度有所提升。其中 GCN+LSTM+DTW 和 GCN+LSTM 以及 GCN+LSTM+ED 的预测值和真实值对比如图 4-8。

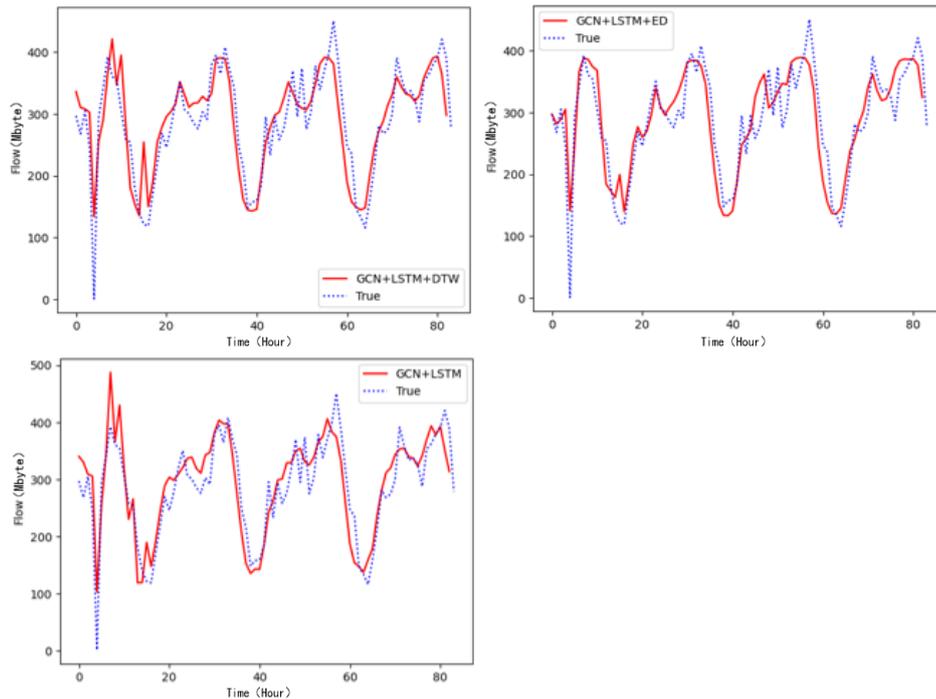


图 4-8 DTW 相似度对比结果
Figure 4-8 DTW Similarity Comparison Result

GCN+LSTM+DTW 使用自动扭曲时间的 DTW 衡量两个时序数据相似度。可知在 GCN 提取空间特征的基础上，使用 DTW 进一步提取节点空间上的相似度是有效的。且该结果证明能自动扭曲时间计算相似距离的 DTW 能在复杂网络中，比计算时序的欧氏距离更有效的预测流量。

使用不同的基准进行对比后，本文得的结果：捕捉空间关系有助于流量预测；物理距离不能有效衡量两个节点流量之间的相似性；使用原始时间轴计算流量相似性缺乏有效性，但是使用扭曲时间轴的 DTW 计算相似距离是有效的。

4.3 划分拓扑子网络

如 2.4 节所述，传输网拓扑带有子网络定义，需要按定义将整个拓扑划分为数个子网络，为后文针对负载均衡的拓扑优化和指标评测提供基础。

传输网拓扑在设计之初就具有层次，高容量节点被设计在具有汇聚作用的主要链路上，而低容量节点会被设计在次级链路上。并且次级链路都会挂在具有汇聚作用的主链路上。这样的设计使得整个拓扑网络可以划分成多个由汇聚链路作为主要链路，次要接入链路作为副链路子网络。本文依据 2.4.2 小节的子网络定义，开始划分网络为子网络。首先，使用堆栈式深度优先遍历寻找主副链路集合，根据主副链路是否相交以及链路长短区分主副链路。然后，使用递归函数去除副链路冗

余节点,并区分了副链路中的主次链路。接着,使用集合运算寻找下挂链路。最后,使用找到了主链路、副链路以及下挂链路组成了所需要的子网络。下面介绍划分子网络的算法。

(1)堆栈式深度优先遍历寻找链路集合

深度优先遍历的本质是优先访问当前节点的第一个邻居节点。其有两种实现方式:递归式与堆栈式深度优先遍历。考虑到本课题所研究的拓扑图约束条件较多且图中节点数量庞大,使用递归式深度优先会出现递归调用过多导致内存占用大、时间复杂度过高问题,本文选择使用堆栈式深度优先算法。具体算法如 4-3 所示。

表 4-3 深度优先链路搜寻算法
Table 4-3 Depth-first Traversal Search Link

Algorithm 1: Depth-first traversal search link

Input: Adjacency matrix G , Node N , Node type T , Start node S , End node E

Output: Paths P

```

Initialize Stack, Instack
1 Stack[0] ← S, Instack[S] ← 1
2 while Stack > 0 do
3   top element ← Stack[-1]
4   if top element ≠ E then
5     for each index in [c, n] do
6       j ← index, type ← T[index]
7       if not visited this node ∧ (type = previous type ∧ Stack = 1) then
8         Instack[index] ← 1, Stack[-1] ← N[index]
9         search for the next node
10        break
11       if Search to the end then
12         top element ← Stack[-1], Instack[top element index] ← 0
13         pop(Stack)
14         continue the search of the previous node
15     if top element = end then
16       P ← Stack
17       pop(Stack)
18 Return P

```

本课题使用堆栈式深度优先遍历算法时,输入两个节点,一个节点代表遍历的起始节点,也是链路的起始点,一个节点代表链路的终点。首尾节点类型依据子网络定义需要是 G 类型或 H 类型。深度优先遍历的目的就是找出两点之间符合主

副链路定义的所有链路。因为需要找出的是按定义划分的链路，且链路节点之间存在约束条件，所以该深度优先遍历与传统的深度优先遍历相比，在遍历节点的时需通过设置堆栈的入栈条件，保证所搜寻链路的相邻节点之间满足约束条件。

使用该算法寻找主副链路时，算法的输出结果是两点间的所有链路，这些链路中必然包含主链路和副链路。但需要注意的是该算法找出的链路无法区分主副链路。根据子网络定义，主链路要求中间节点的容量值大小相同，可以根据这个条件筛选出主链路候选集合。候选链路集合中有主链路和中间节点容量大小相同的副链路，根据主网络定义，有交集的链路选择较长的作为主链路，其余为副链路。至此区分出了主副链路。

(2) 递归函数去除冗余

深度优先遍历搜寻到的副链路包含主链路节点，而冗余节点不便于计算。为了后面处理拓扑网络以及计算方便。需要把副链路的存储结构中不属于副链路的主链路部分剔除。去除冗余节点无法使用区分主副链路交集去重思想，这是因为副链路之间也是有层次结构的，一条副链路可能下挂在另一条副链路下，而现在的副链路集合中是没有副链路之间的从属信息的，从而无法判断那条链路中的节点是冗余节点。考虑到这些问题，本文设计了一个递归算法来去除冗余。具体算法如表 4-4 所示。

表 4-4 递归去冗余算法
Table 4-4 Recursive De-redundancy Algorithm

Algorithm 2: Depth-first traversal search link

Input: Sub paths set P_s , Redundant node set S_r

Output: Non redundant sub paths set P_n

*Initialize shortest path p_s , non redundant path p_n , this recursion return r_n ,
next recursion return r_c*

- 1 **for** node in p_s **do**
- 2 **if** (node $\cap S_r$) **then**
- 3 **Continue**
- 4 add node in S_r
- 5 append node in p_n
- 6 **if** $P_s = 0$ **then**
- 7 **return** $r_n \leftarrow r_c$
- 8 **if** $P_s \neq 0$ **then**
- 9 $r_c \leftarrow \text{callself}(P_s, S_r)$
- 10 **Return** $r_c + r_n$

该算法是将副链路集合送入一个递归函数，该函数的每层递归输出为当前副链路集合中最短的链路，且要求该链路节点不在冗余节点集合中。递归的临界条件就是副链路集合长度为 0，递归等式为：

$$S_i = s_i + S_{i-1} \quad (4-12)$$

其中 S_i 为第 i 层递归中的副链路集合， S_{i-1} 为第 $i-1$ 层递归中的副链路集合， s_i 为第 i 层递归中链路集合里的最短链路。这个递归函数每次都找出集合中最短的一条，而最短的一条在逻辑上被视为副链路中的主要副链路，其余较长的链路被视为挂在副链路下的链路。在每层递归返回之前通过冗余节点集合来判断该最短链路中的冗余点，从而去除冗余。最终，通过这个递归函数得到了一个没有冗余节点的副链路集合。

(3)集合运算寻找下挂链路

将主链路和副链路的所有节点加入到主副节点集合中，再使用拓扑数据获得所有节点的集合，所有节点集合与主副节点集合的差为下挂节点集合。

现在有了下挂节点集合，需要确认每条下挂链路的从属的子网络。从下挂链路定义可知，下挂链路的某个节点一定是连接着主链路节点或者副链路节点的，这意味着下挂链路两端的首尾节点当中，一定有一个节点连接到了不在下挂节点集合中的点。通过检查每个下挂链路节点的连接关系，确认是否连接到了下挂链路节点集合外的点上来找出下挂链路首个节点。找出了首节点及其所连接的点，查看所连接点的子网络属性，最终确认了下挂链路所在子网络的编号。至此找到了主链路、副链路和下挂链路。主链路和连接在主链路下的副链路和下挂链路一起组成了子网络。

4.4 随机让渡节点的算法

本节介绍了拓扑优化算法的实现，解释了遗传算法如何在不打破约束条件的情况下让渡节点。并且使用上文所预测的流量数据、划分好的子网络进行拓扑优化，最后展示了实验结果。

4.4.1 节点分层分类

为了使遗传算法在随机变动时，不打破每个节点只属于一个子网络的约束条件，同时也保证节点不会连接到距离限制外的点和不会连接到小容量点，需要将随机变动的方式从随机连边变为随机让渡节点，在每次让渡节点时都将原有连接

关系继承给等效节点。为了辅助让渡节点本文提出了节点分层分类。节点分层分类是将所有节点按照其容量大小、所属子网络划分成数个集合，每个集合中的点都有相同容量、属于同一子网络。这样做的好处是集合中的所有点都可以视为等效点。当集合中的一个点发生变动时，可以将变动点的所有连边继承给等效点，从而维持随机变动点的邻接节点所属子网络。

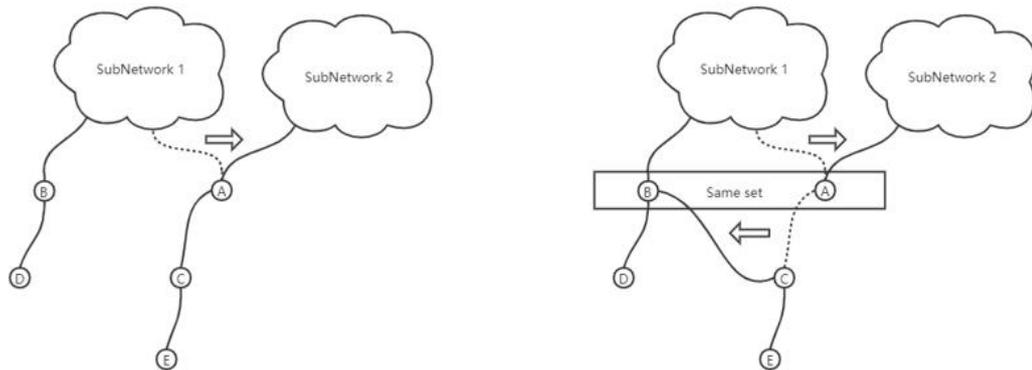


图 4-9 节点分层分类

Figure 4-9 Node Divided By Capacity and Subnet

如图 4-9 所示，在分层分类中节点 A、节点 B 有相同容量，且在节点 A 变动前，节点 A 和节点 B 同属一个子网络，所以两个节点属于同一个集合，互为等效点。在节点 A 进行变动时，将原有的所有连接关系让渡给等效节点 B。这样节点 C 和节点 E 依然保持了原有的从属子网络，且符合上下节点容量大小约束。为了简化遗传算法的代码编写，本文设计了基于分层分类的节点变动算法，算法流程如图 4-10 所示。

算法输入为需要变动的节点、变动目标子网络以及等效点集合。算法流程的第一步是判断变动节点能否连接到目标子网络中节点，设置这一步的原因是后文中的遗传算法只负责找出目标子网络，不进行可连接性的检查。接着查看变动点的等效点集合，查看是否有除去自身的等效点，如果有等效点，顺序选取等效点作为变动点的替代点。等效点能否连接到变动点的邻接节点也是需要检查的，这是因为分层分类只针对了容量大小和从属子网络，没有考虑距离限制。没有考虑距离限制的原因是在节点集合中，如果再按距离划分集合的话，会出现大量元素个数只有 1 的集合，违反了分层分类的设计初衷。算法最后判断邻接点是否可以连接到替代点，如果可以连接，将邻接点连接然后将变动点连接到目标子网络的节点上，完成整个算法的流程。

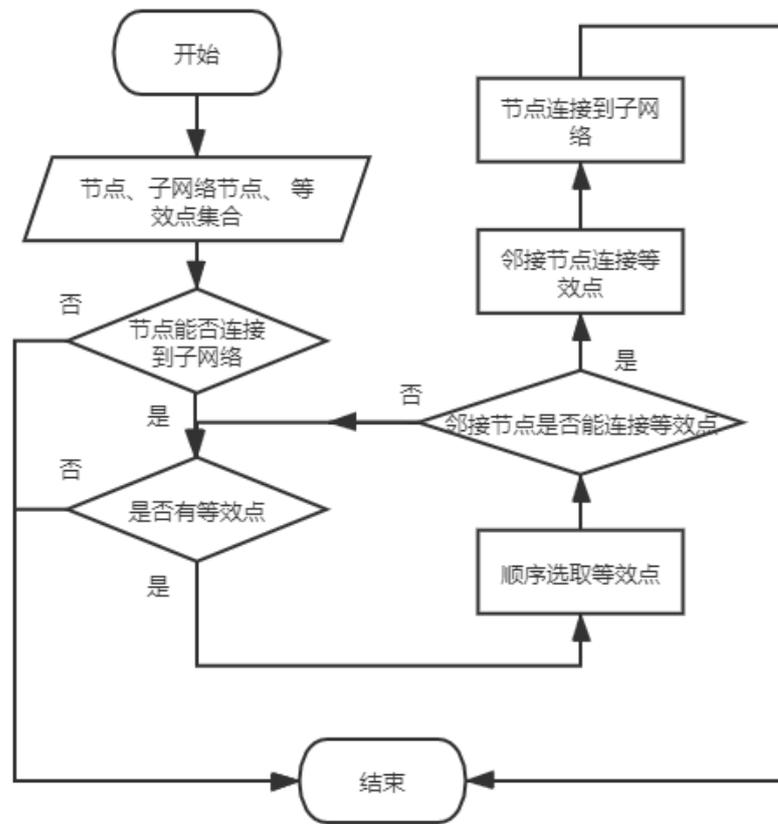


图 4-10 节点变动算法
Figure 4-10 Node Transfer Algorithm

4.4.2 随机让渡节点的遗传算法实现

拓扑网络优化结果的好坏需要通过各个子网络的容量利用率的方差来衡量。接下来介绍优化算法的负载均衡指标，并且该指标也是遗传算法的适应度函数。

使用上文中划分好的子网络作为评测单元进行拓扑优化。需要衡量拓扑优化后的每个小时子网络容量利用率：

$$f_p^{(l)} = \frac{\sum_i^N f_i^{(l)}}{A} \quad (4-13)$$

式(4-13)中 $f_{x_i}^{(l)}$ 是节点 i 在 l 时刻的流量， N 是子网络中节点数目， A 是子网络的容量值，该式衡量了在 l 时刻子网络容量的使用情况。

为了负载均衡还需要计算整个拓扑网络的平均容量利用率：

$$\mu^{(l)} = \frac{\sum_i^M f_{p_i}^{(l)}}{M} \quad (4-14)$$

其中的 $f_{p_i}^{(l)}$ 为 l 时刻 i 子网络的容量利用率， M 为子网络个数。有了平均值就可以确

定各个子网络的容量利用率距离平均值的偏离程度。一个优化良好的拓扑网络，需要各个子网中不会有过多低负载节点也不会有太多高负载节点。从而使容量利用率维持在一个不高不低的程度，在充分利用网络资源的同时，保持一定抗风险能力。衡量偏离程度需要计算方差：

$$E^{(l)} = \frac{\sum_i^M (f_{p_i}^{(l)} - \mu^{(l)})^2}{M} \quad (4-15)$$

其中中的 $E^{(l)}$ 为 l 时刻所有子网络容量利用率 $f_{p_i}^{(l)}$ 的方差， M 为子网络数目。

最终负载均衡指标为：

$$L = ratio - (E_{avg} + E_{min} + E_{max}) \quad (4-16)$$

其中中的 $ratio$ 是所有子网络容量利用率 f_p 达到目标范围 μ 上下波动 2/5 范围的子网络数目的每小时均值。 $ratio$ 反映了网络拓扑负载均衡程度，而式(4-16)中的 E_{avg} 、 E_{min} 和 E_{max} 分别是所有子网络容量利用率方差的每小时均值、方差的每小时最小值和方差的每小时最大值，代表了优化后的网络拓扑负载不均衡程度。使用式(4-16)可以综合衡量拓扑网络每天的负载均衡性。

为了提高指标 L ，算法通过交换子网络中的节点，改变每个子网络的容量利用率，最终影响拓扑网络的负载均衡。即负载均衡的拓扑优化的实质上就是优化各个子网络中的点的构成。所以算法的目标就是要找到各个子网络节点的最佳构成。将所有非主链路上的接入节点按容量大小排序，然后将这些节点的所属子网络编号作为遗传编码 X ：

$$X = \{x_1, x_2, \dots, x_N\} \quad (4-17)$$

式(4-17)中的 N 为非主链路上的接入节点个数， x 为节点所属子网络编号。在算法迭代中对该编码进行变异、交叉以及选择，期望获得能使网络负载均衡最优的编码。

在设置了适应度函数和编码后，可以开始运行遗传算法来优化网络拓扑。本文的遗传算法使用了自适应遗传算法的种群消滅方法和自适应变异算子，种群消滅保证了遗传算法的并行搜索效率，自适应变异算子保证了遗传算法能有更大机率将优良个体的模式传递下去。算法的具体流程如 4-11 所示。

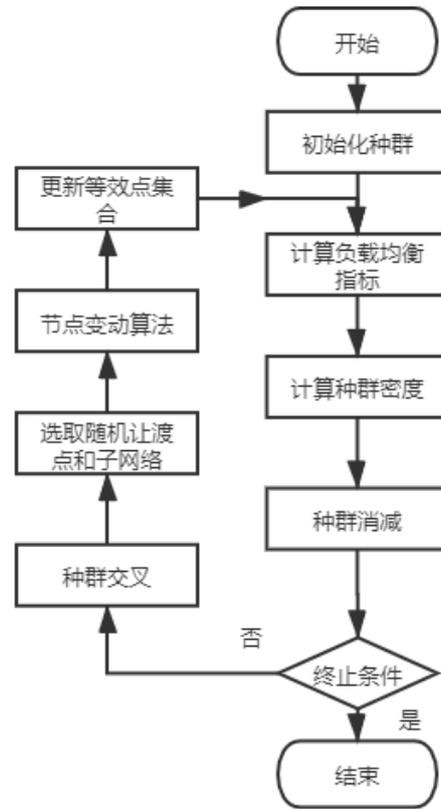


图 4-11 遗传算法

Figure 4-11 Genetic Algorithm

遗传算法初始种群非常关键。因为遗传算法中的每个个体都可以视为解空间中的一点，为了尽可能的探索更多的解空间，进而快速收敛，本文使用自适应遗传算法中的种群消减方法。种群消减需要初始化的种群均匀分散在空间中，这样每个个体进行随机探索时，其探索区域才不会出现覆盖。为此需要衡量两个个体间的距离：

$$D = \sum_{i=1}^N A(x_i^{(1)}, x_i^{(2)}) \quad (4-18)$$

$$A(x_i^{(1)}, x_i^{(2)}) = \begin{cases} 1, & \text{for}(x_i^{(1)} = x_i^{(2)}) \\ 0, & \text{for}(x_i^{(1)} \neq x_i^{(2)}) \end{cases} \quad (4-19)$$

其中 $x_i^{(1)}$ 与 $x_i^{(2)}$ 分别为遗传编码 $X^{(1)}$, $X^{(2)}$ 的第 i 节点所对应子网络编号， A 函数用于比较两个遗传因子是否相同， D 代表了两个遗传编码的距离。对于初始化种群中的个体计算距离后，设置一个阈值，当两个个体的距离高于阈值意味着两个个体过于接近，此时计算两个个体的负载均衡指标 L ，抛弃指标小的个体。每抛弃一个个体就再初始化一个新个体，直到达到要求种群数目。使用这种手段可以保证初始种群一定程度分散在解空间中。

初始化完成后，进入算法流程。对于每个个体计算其负载均衡指标 L ，然后计

算个体与个体之间的距离 D ,进行种群消减,消减的逻辑和种群初始化一致,消除高于阈值个体对中的负载均衡指标低的个体。接着开始判断终止条件,本文的终止条件设置为迭代次数。当不满足终止条件时,开始种群交叉生成新的个体,种群交叉的方法就是将两个个体的编码 X 进行单点交叉。交叉的次数是由种群消减时消减数决定的,交叉时需要保持种群数量。最后随机选取个体进行变异,注意为了保证优秀个体能将负载均衡性良好的子网构成遗传到下一代,需要在每次交叉时,使用自适应变异算子来决定个体是否变异,设置变异率与拓扑网络负载均衡指标成反比,以保证优良个体有更大机率遗传到下一代,自适应变异算子定义为:

$$P_m = \begin{cases} \alpha(0.5 - L_{nml}^2), 0 \leq L_{nml} \leq 0.5 \\ \alpha(1 - L_{nml})^2, 0.5 \leq L_{nml} \leq 1 \end{cases} \quad (4-20)$$

其中 P_m 代表 m 个体的变异率, L_{nml} 代表个体 m 适应度值 L 的归一化值, α 为变异系数。对于被选中的个体,变异方法就是在编码值随机选取数个节点,再随机选取目标子网络。将变动节点和目标子网络送入节点变动算法。需要注意的是,节点变动算法是有可能失败的,原因为等效点不足和邻居节点和等效点距离过大。这种情况视为不进行变异,对后续算法流程无影响。最后就是根据交叉和变异中的变动节点算法失败与否,来更新等效点集合。

4.4.3 拓扑优化结果与分析

本文成功在传输网拓扑上使用遗传算法,并将预测流量输入算法,最终实现了基于时空预测的网络拓扑优化系统。在研究和实现系统的过程中主要关注以下三个问题:

(1)遗传算法在本课题研究的拓扑网络中是否有效

因为拓扑网络优化问题是一个 NP-hard 问题,所以难以获得梯度信息,常选择使用遗传算法进行全局搜索,需要验证遗传算法在本文研究的传输网拓扑结构上的有效性。

(2)预测精准度对网络优化的影响

设计的系统需要考虑节点流量未来的变化趋势。在拓扑优化时,需要验证适应度函数使用的预测流量数据越准确,优化后的网络拓扑性能是否越接近真实流量数据所优化网络的性能。

(3)基于时空预测的拓扑优化系统的有效性

需要验证整个系统在使用历史数据和未来预测数据作为适应度函数输入时,系统所优化拓扑的负载均衡性能相比较原拓扑的负载均衡性能是否有提升,以

此证明系统有效性。

本文针对以上问题设计实验：

(1) 遗传算法在本课题研究的拓扑网络中是否有效

本文提出的随机让渡节点的遗传算法的主要目的是在具有约束条件的网络拓扑上充分探索解空间。本文设置基准为启发式节点让渡算法，基准算法因设计缺陷会出现过早陷入局部最优，不能充分探索解空间的情况。本文通过实验对照说明随机让渡节点的遗传算法的有效性。

在实验中，遗传算法的种群大小选择为 200，自适应变异算子的系数为 0.6，距离阈值为 20，使用历史流量作为适应度函数输入。分别使用启发式节点让渡算法和随机让渡节点的遗传算法优化拓扑网络。其结果如表 4-5 所示。

表 4-5 遗传算法基准对比
Table 4-5 Genetic Algorithm Baseline Comparison

算法	L
原拓扑	36.813
启发式节点让渡算法	37.256
随机让渡节点的遗传算法	40.395

算法的收敛情况如图 4-12 所示。结果表明随机让渡节点的遗传算法的有效性，对比基准负载均衡指标提升了 8.4%。

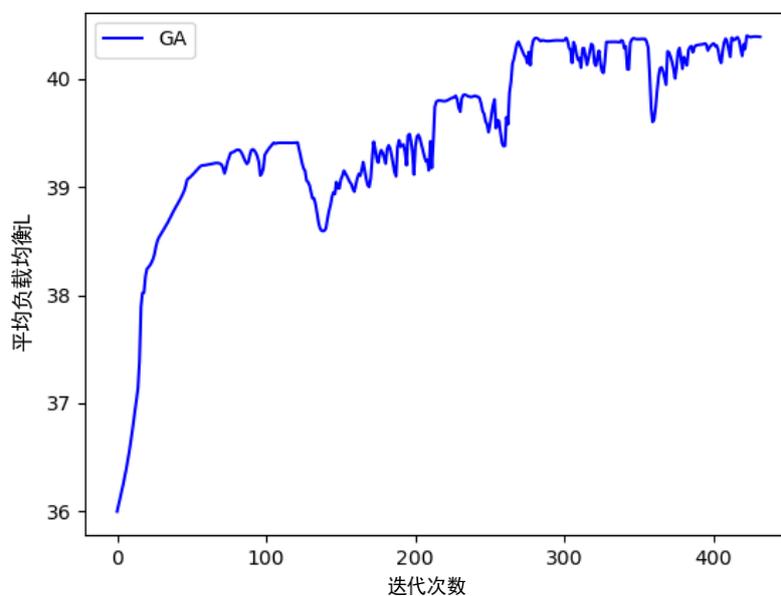


图 4-12 遗传算法收敛情况
Figure 4-12 Convergence of Genetic Algorithm

为了证明随机让渡节点的遗传算法在探索解空间上的优越性，需要说明启发

式节点让渡算法的缺陷。基准算法首先按照容量利用率将子网络排序得到高负载子网络和低负载子网络，循环执行将高负载子网络的高流量节点依据约束条件让渡到低负载子网络。该算法面临一个问题：受到距离限制，高流量节点不会总是让渡到负载最低子网络，而是会让渡到附近的子网络中，而附近的子网络在连续接受几次高负载子网络中节点后会变成最高负载子网络，新的最高负载子网络又开始让渡曾经接受果的高流量节点。这就导致整个算法实质上一直在周期性的交换数个相同的高流量节点。而本文采用的遗传算法，其本质是在解空间中的随机探索，每次节点的变动都没有针对性，很难出现如基准节点让渡算法一样的陷入局部最优问题。

(2) 预测精准度对网络优化的影响

接着，为了说明时空预测对于拓扑优化的有效性，我们分别使用真实流量、叠加高斯噪音的真实流量、LSTM 的预测流量、GCN+LSTM 的预测流量以及 GCN+LSTM+DTW 的预测流量作为适应度函数输入的数据，使用遗传算法优化拓扑，对得到的四个拓扑网络进行评估。评估的指标为：

$$d = \frac{|L_p - L_h|}{L_h} \quad (4-21)$$

其中 L_p 为预测流量或加噪声流量所优化的拓扑在承载真实流量数据时的负载均衡指标， L_h 为真实流量优化的拓扑在承载真实流量时的负载均衡指标。在这个对比实验中，注重的是使用预测流量优化的网络拓扑和真实流量所优化的拓扑的差别，期望越精准的预测，即 d 越小越好。其优化结果如表 4-6 所示。

表 4-6 各预测模型性能对比
Table 4-6 Performance comparison of various prediction models

不同数据优化的拓扑	L_p	d
真实数据所优化拓扑 L_h	40.396	0%
50 噪音真实数据所优化拓扑	35.648	11.75%
LSTM 所优化拓扑	37.665	6.76%
GCN+LSTM+ED 所优化拓扑	38.035	5.84%
GCN+LSTM+DTW 所优化拓扑	38.932	3.62%

加了波动范围 50 的高斯噪音流量所优化拓扑其指标和基准相差 11.75%，LSTM 所优化拓扑与基准对比相差 6.76%，GCN+LSTM+ED 所优化拓扑与基准对比相差 5.84%，GCN+LSTM+DTW 所优化拓扑与基准对比相差 3.62%。说明各个

结果与流量预测的精准度有关，流量预测越准确，优化的拓扑网络在现实中越有效。

并且为了说明 DTW 的有效性，对比 GCN+LSTM+DTW 所优化拓扑与 GCN+LSTM+ED 所优化拓扑，相差减少 2.22%。

(3) 基于时空预测的拓扑优化系统的有效性

系统设计目的在于使得优化后的拓扑结构在现有节点流量的未来变化基础上，满足负载均衡。将历史数据和 GCN+LSTM+DTW 预测的流量数据作为优化目标函数的输入，即适应度函数的输入，优化拓扑。优化后的拓扑与优化前的拓扑的对比结果如表 4-7 所示。

表 4-7 基于时空预测的拓扑优化系统性
Table 4-6 Topology optimization system performance based on time-space prediction

拓扑网络	L
优化前拓扑	35.981
优化后拓扑	38.520

优化前的部分拓扑如图 4-13，该拓扑在承载历史流量和预测流量时的负载均衡指标为 35.981。

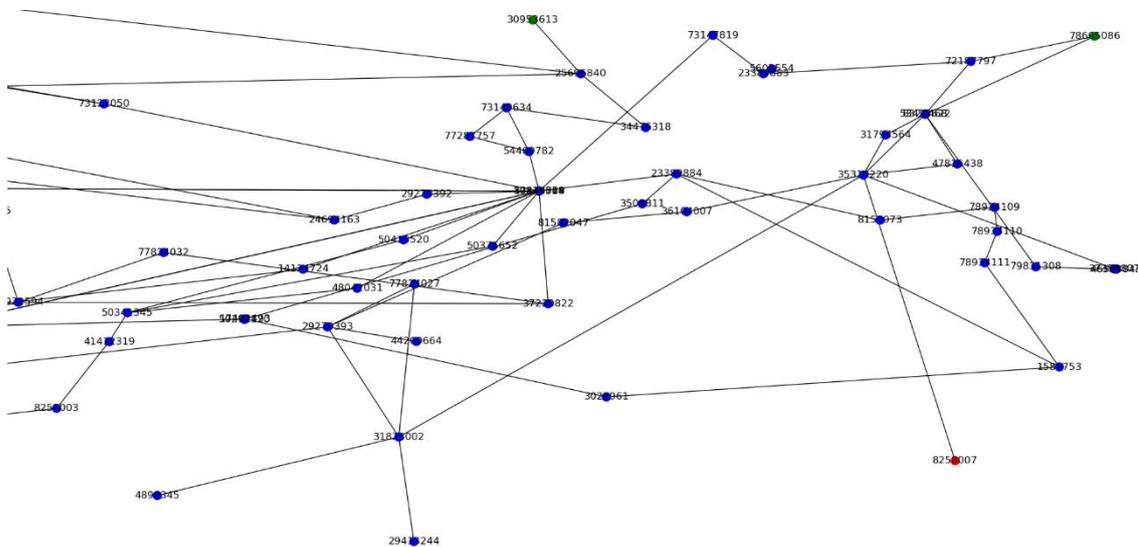


图 4-13 优化前网络拓扑
Figure 4-13 Network Topology Before Optimization

在使用历史数据和预测数据作为适应度函数输入所优化的拓扑中，其负载指标为 38.520，优化后的部分拓扑图如 4-14 所示。优化后的部分拓扑将 2 个用绿色表示的小流量节点 30953613 和节点 78665086 让渡给了高负载网络，并且将 1 个用红色表示的高负载节点 8258007 让渡给了低负载网络。相比于无优化原拓扑的负载均衡指标，其负载均衡性能提升了 7.05%。结果证明了本文所设计系统的有效

性，成功的在考虑流量变化趋势下优化了网络拓扑，改善了网络的负载均衡性。

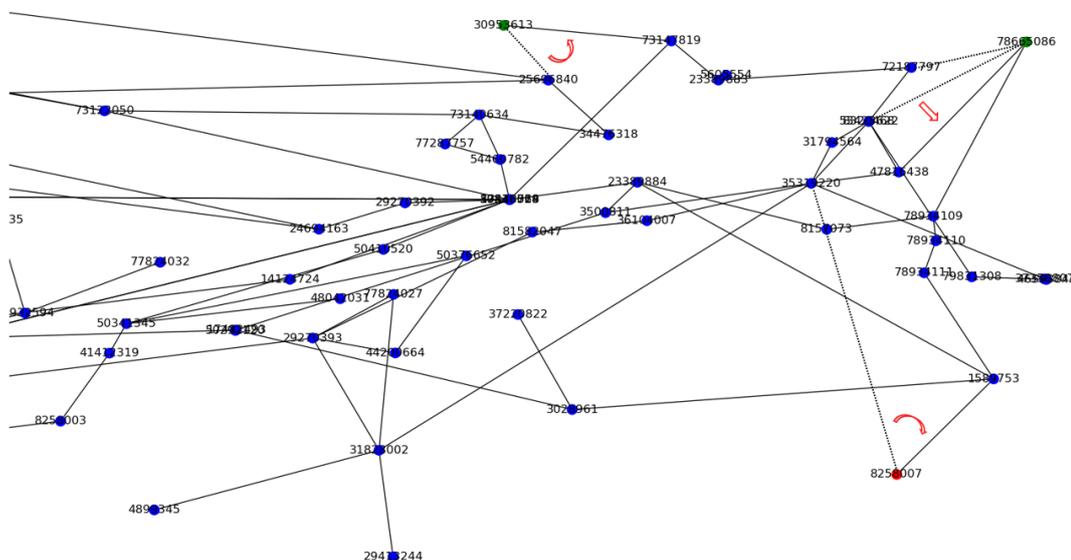


图 4-14 优化后网络拓扑
Figure 4-14 Optimized Network Topology

4.5 本章小结

本章主要介绍了基于时空预测的 5G 拓扑网络优化系统的实现和实验结果。与现有的 GCN+LSTM 预测框架比起来，本文添加 DTW 模块，解决了流量全局相似局部相异的问题，自动规整时间将相似数据对齐，准确衡量了流量时序相似性，提供了更可靠的空间信息，进而更准确地预测了数据。

接着为了更有效的探索拓扑解空间，本文解决了遗传算法在约束条件下无法使用的问题。通过改变遗传算法的随机探索模式和将网络节点分层分类，将遗传算法运用到拓扑网络中，充分地探索了解空间。

对时空预测模型进行了实验验证，实验结果证实 DTW 模块提高了时空预测准确度，相比较于使用两个时间轴对应的欧氏距离相似度的 GCN+LSTM+ED，使用了 DTW 的模型 MAE 提升了 3.7%，RMSE 提升了 2.3%。然后对分层分类遗传算法进行了实验，实验结果证实随机让渡节点的遗传算法在拓扑解空间探索的有效性，相比较基准启发式让渡节点算法，随机让渡节点的遗传算法的负载均衡指标提高了 8.4%。同时为了验证流量预测对系统性能的影响，本文衡量了不同预测数据作为适应度函数输入时所优化的不同拓扑的性能，结果显示 GCN+LSTM+DTW 所优化的拓扑网络与历史数据所优化的拓扑网络性能更相似，性能比达到了 96.375%，证明了预测越精准，优化的拓扑在真实环境中越有效。最后为了验证整个系统的有

效性，将历史流量数据和预测流量数据送入适应度函数，运行遗传算法，优化后的拓扑相比较原拓扑，负载均衡指标提高 7.05%，证明了基于时空预测的网络拓扑优化系统的有效性。

5 结论

5.1 本文工作总结

本文为了解决现有传输网拓扑结构在 5G 时代的负载不均衡问题,设计了一个基于时空预测的网络拓扑优化系统。系统通过图卷积神经网络捕获空间信息,利用长短期记忆网络捕获时间信息,联合时空信息进行流量预测。系统以预测流量和历史流量作为拓扑优化算法中的优化目标函数输入,使用遗传算法优化网络拓扑。在工作过程中主要面临以下难点: 1)本文的目的是通过获取空间信息辅助时间预测,但是数据集中缺乏有效的空间信息,需要挖掘信息以提高预测准度。在工作中发现流量之间出现全局相似、局部相异的特点,为了准确预测流量需要考虑该情况。2)拓扑网络节点以及连边数量庞大。在进行拓扑优化时,难以计算和处理整个拓扑网络,需要将整个拓扑网络划分成数个子网络。3)传统的遗传算法对拓扑图中的连边进行编码,算法使用编码来随机探索节点之间的连接方式。但由于传输网节点之间的连接存在约束条件,导致传统遗传算法的随机连边会违反节点之间的约束条件。

本文主要工作如下:

(1)本文提出在传输网络中使用时空预测,并且对节点流量之间的相似性进行了研究。在研究过程中发现高度相似的两个流量时序之间存在不规则的时间错位,表现为相邻节点的时序波形全局相似、局部相异。使用时间对应的欧式相似距离难以正确衡量波形间的相似性。为此,本文在时空预测网络中使用动态时间规整来衡量流量时序的相似度。动态时间规整很好地解决了时序数据之间存在全局相似、局部相异的问题,进而准确衡量了两个流量的相似性,为时空预测提供了有效的空间信息。最终结果显示相比较于非时空预测基准模型,使用的动态时间规整相似距离的时空预测模型在预测精度上提升了 6.7%。

(2)本文研究的传输拓扑网络节点和连边数目庞大。考虑到传输网的拓扑结构特点和节点属性,设计了用于搜寻链路的堆栈式深度优先遍历算法和过滤节点的递归式去冗余算法,使用算法将整个拓扑网划分成数个除了汇聚节点外没有公共节点的子网络。为后序拓扑优化算法提供了基础。

(3)针对传输网节点之间的连接约束,改进了遗传算法,将传统遗传算法的随机探索模式从随机连边改为随机让渡节点。同时设计了分层分类等效点集合,使用等效点集合辅助遗传算法的节点让渡。成功地在传输网络上使用遗传算法。实现了基于时空预测的网络拓扑优化系统。在实验中,对比了优化后的拓扑网络和未优化

的原拓扑网络，优化后的拓扑网络负载均衡指标提升了 7.05%，证明了基于时空预测的网络拓扑优化系统的有效性。

5.2 未来工作展望

本文的工作为实现基于时空预测的网络拓扑优化系统，系统功能主要分为流量预测和拓扑优化。流量预测面对的问题是空间信息挖掘不充分。拓扑优化面临的问题是缺乏针对传输网约束条件的非启发式优化算法。本文希望未来的工作能在以下方面进行突破：本文流量预测使用的空间信息仅考虑了流量相似度，而为了准确获得空间信息需要考虑多因素影响，未来计划捕获多维度空间信息，并融合多维度信息作为时空预测的输入。在拓扑优化工作方面，本文使用的遗传算法适用度高，但与一些针对具体问题具体设计的优化算法相比会缺乏优势，未来需要设计针对传输网络的非启发式拓扑优化算法。

参考文献

- [1] Saad A, Khan S A, Mahmood A. A multi-objective evolutionary artificial bee colony algorithm for optimizing network topology design[J]. *Swarm and Evolutionary Computation*, 2018, 38: 187-201.
- [2] Liu B, Iwamura K. Topological optimization models for communication network with multiple reliability goals[J]. *Computers & Mathematics with Applications*, 2000, 39(7-8): 59-69.
- [3] 马永杰,云文霞.遗传算法研究进展[J].*计算机应用研究*,2012,29(04):1201-1206+1210.
- [4] 王洪鹏,李建强.5G 网络大带宽驱动下传输网拓扑结构优化模型的研究[J].*通信电源技术*,2020,37(06):176-177.
- [5] 康梦轩,宋俊平,范鹏飞,高博文,周旭,李琢.基于深度学习的网络流量预测研究综述[J].*计算机工程与应用*,.doi:10.3778/j.issn.1002-8331.2101-0402.
- [6] 黎维,陶蔚,周星宇,潘志松.时空序列预测方法综述[J].*计算机应用研究*,2020,37(10):2881-2888.
- [7] Chai D, Wang L, Yang Q. Bike flow prediction with multi-graph convolutional networks[C]//*Proceedings of the 26th ACM SIGSPATIAL international conference on advances in geographic information systems*. 2018: 397-400.
- [8] Yang G, Wang Y, Yu H, et al. Short-term traffic state prediction based on the spatiotemporal features of critical road sections[J]. *Sensors*, 2018, 18(7): 2287.
- [9] Zhang X, Huang C, Xu Y, et al. Spatial-Temporal Convolutional Graph Attention Networks for Citywide Traffic Flow Forecasting[C]//*Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2020: 1853-1862.
- [10] Yao H, Tang X, Wei H, et al. Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction[C]//*Proceedings of the AAAI conference on artificial intelligence*. 2019, 33(01): 5668-5675.
- [11] Zhang J, Zheng Y, Sun J, et al. Flow prediction in spatio-temporal networks based on multitask deep learning[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2019, 32(3): 468-478.
- [12] Sun S, Wu H, Xiang L. City-wide traffic flow forecasting using a deep convolutional neural network[J]. *Sensors*, 2020, 20(2): 421.
- [13] Li M, Wang Y, Wang Z, et al. A deep learning method based on an attention mechanism for wireless network traffic prediction[J]. *Ad Hoc Networks*, 2020, 107: 102258.
- [14] Guo S, Lin Y, Feng N, et al. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting[C]//*Proceedings of the AAAI Conference on Artificial Intelligence*. 2019, 33(01): 922-929.
- [15] Yao H, Wu F, Ke J, et al. Deep multi-view spatial-temporal network for taxi demand prediction[C]//*Proceedings of the AAAI Conference on Artificial Intelligence*. 2018, 32(1).
- [16] Chen J, Xing H, Yang H, et al. Network traffic prediction based on LSTM networks with genetic algorithm[C]//*International Conference On Signal And Information Processing*,

- Networking And Computers. Springer, Singapore, 2018: 411-419.
- [17] Chen Y, Shu L, Wang L. Traffic flow prediction with big data: A deep learning based time series model[C]//2017 IEEE conference on computer communications workshops (INFOCOM WKSHPs). IEEE, 2017: 1010-1011.
- [18] Liu Z, Wang Z, Yin X, et al. Traffic Matrix Prediction Based on Deep Learning for Dynamic Traffic Engineering[C]//2019 IEEE Symposium on Computers and Communications (ISCC). IEEE, 2019: 1-7.
- [19] Lu H, Yang F. A network traffic prediction model based on wavelet transformation and LSTM network[C]//2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS). IEEE, 2018: 1-4.
- [20] Hua Y, Zhao Z, Li R, et al. Deep learning with long short-term memory for time series prediction[J]. IEEE Communications Magazine, 2019, 57(6): 114-119.
- [21] He Y, Hui C W. A binary coding genetic algorithm for multi-purpose process scheduling: A case study[J]. Chemical engineering science, 2010, 65(16): 4816-4828.
- [22] 陈辉,张家树,张超.实数编码混沌量子遗传算法[J].控制与决策,2005(11):102-105.
- [23] Tang K Z, Sun T K, Yang J Y. An improved genetic algorithm based on a novel selection strategy for nonlinear programming problems[J]. Computers & Chemical Engineering, 2011, 35(4): 615-621.
- [24] Dengiz B, Altiparmak F, Smith A E. Local search genetic algorithm for optimal design of reliable networks[J]. IEEE transactions on Evolutionary Computation, 1997, 1(3): 179-188.
- [25] Adalsteinsson D, Sethian J A. A fast level set method for propagating interfaces[J]. Journal of computational physics, 1995, 118(2): 269-277.
- [26] Caselles V, Catté F, Coll T, et al. A geometric model for active contours in image processing[J]. Numerische mathematik, 1993, 66(1): 1-31.
- [27] Bernard O, Friboulet D, Thévenaz P, et al. Variational B-spline level-set: a linear filtering approach for fast deformable model evolution[J]. IEEE Transactions on Image Processing, 2009, 18(6): 1179-1191.
- [28] Cremers D, Tischhäuser F, Weickert J, et al. Diffusion snakes: Introducing statistical shape knowledge into the Mumford-Shah functional[J]. International journal of computer vision, 2002, 50(3): 295-313.
- [29] 刘建磊,冯大政.一种基于二维拉格朗日连续水平集的图像分割方法[J].电子与信息学报,2010,32(07):1712-1716.
- [30] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. IEEE transactions on evolutionary computation, 2002, 6(2): 182-197.
- [31] Irani R, Nasimi R. Evolving neural network using real coded genetic algorithm for permeability estimation of the reservoir[J]. Expert Systems with Applications, 2011, 38(8): 9862-9866.
- [32] Liu F, Zeng G. Study of genetic algorithm with reinforcement learning to solve the TSP[J]. Expert Systems with Applications, 2009, 36(3): 6995-7001.
- [33] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.

- [34] Kalchbrenner, N., Grefenstette, E., & Blunsom, P. (2014). A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*.
- [35] Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
- [36] Hammond D K, Vandergheynst P, Gribonval R. Wavelets on graphs via spectral graph theory[J]. *Applied and Computational Harmonic Analysis*, 2011, 30(2): 129-150.
- [37] Defferrard, M., Bresson, X., & Vandergheynst, P. (2016). Convolutional neural networks on graphs with fast localized spectral filtering. *arXiv preprint arXiv:1606.09375*.
- [38] Busetti F. Genetic algorithms overview[J]. Retrieved on December, 2007, 1.
- [39] 李建忠.利用 PTN 通信环网,实现高可靠性信息网络[J].*硅谷*,2013,6(08):80-81.
- [40] 卢研平,吕光宏.双归属环形网络拓扑规划算法设计[J].*电信科学*,2012,28(8):69-74.
- [41] Muda, L., Begam, M., & Elamvazuthi, I. (2010). Voice recognition algorithms using mel frequency cepstral coefficient (MFCC) and dynamic time warping (DTW) techniques. *arXiv preprint arXiv:1003.4083*.