计算机科学
COMPUTER SCIENCE

可靠性感知的边缘计算VNF实例放置

梁晶语, 马博闻, 黄霁崴

引用本文

梁晶语, 马博闻, 黄霁崴. 可靠性感知的边缘计算VNF实例放置[J]. 计算机科学, 2024, 51(6A): 230500064-6.

LIANG Jingyu, MA Bowen, HUANG Jiwei. Reliability-aware VNF Instance Placement in Edge Computing [J]. Computer Science, 2024, 51(6A): 230500064-6.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.230500064

可靠性感知的边缘计算 VNF 实例放置

梁晶语　马博闻　黄霁崴
中国石油大学(北京)石油数据挖掘北京市重点实验室　北京 102249
(2020310701@student.cup.edu.cn)

摘　要　为了解决日益增长的延迟敏感型应用程序和用户需求与计算资源受限的冲突,移动边缘计算(Mobile Edge Computing,MEC)已经成为一种很有前途的计算范式。服务提供商通过在边缘环境中部署虚拟化网络功能(Virtual Network Functions,VNF),为用户提供更加高效和可扩展性的服务供应链(Service Function Chain,SFC)来满足用户请求。若在提供服务过程中出现不可靠的服务或严重的服务失败,可能导致用户的巨大损失,所以网络服务提供商必须保证提供持续可靠的服务。针对该问题,考虑了边缘服务器的可靠性,利用计算统一设备架构(Computational Unified Device Architecture,CUDA)支持的门控循环单元(Gate Recurrent Unite,GRU)来预测 VNF 实例是否可用,通过预测结果,提前对 VNF 进行备份,避免了过度冗余备份造成的成本过高问题。考虑服务器的存储资源有限,提出了基于 VNF 实例可用性的放置(RVP)算法,优化服务提供商的成本。最后对提出的算法进行了性能评估,实验结果验证了 RVP 算法的优越性。

关键词:边缘计算;服务供应链;虚拟化网络功能;可靠性;VNF 实例放置
中图分类号　TP393

Reliability-aware VNF Instance Placement in Edge Computing

LIANG Jingyu,MA Bowen and HUANG Jiwei
Beijing Key Laboratory Petroleum Data Mining,China University of Petroleum-Beijing,Beijing 102249,China

Abstract　Mobile edge computing(MEC) has emerged as a promising computing paradigm to solve the conflict between the growing number of latency-sensitive applications and user demands and the constrained computing resources. To provide users with a more efficient and scalability service function chain(SFC) to satisfy users' requests by deploying virtual network functions (VNF) in the edge environment. Unreliable service or serious service failure in the process of providing service may lead to great loss to users,so the network service provider must ensure the provision of constant and reliable service. Considering the reliability of edge servers for this problem,the gate recurrent unit(GRU) supported by computational unified device architecture(CUDA) is used to predict the availability of VNF,and the VNF instances are backed up in advance through the prediction results,avoiding the problem of excessive cost caused by over-redundant backups. The storage resources of the servers are limited,and VNF instance availability placement(RVP) algorithm is proposed to optimize the cost of service providers. Finally,performance evaluation is performed,and the experimental results show the excellence of the proposed RVP algorithm.

Keywords　Edge computing,Service function chain,Virtual network function,Reliability,VNF instance placement

1　引言

全球数字化的蓬勃发展,以及数据流量的爆炸性增长,大规模新应用的出现以及移动用户对服务质量需求的提升,推动着 MEC 的发展[1]。边缘计算相比云计算来说,更靠近终端设备,并且集中式云计算包涵大量数据的传输和处理,核心网络的负载过高,导致数据传输和处理速率降低[2],而边缘计算通过就近提供计算、网络、智能等关键能力,已逐步成为计算体系的新方向。

大量的边缘智能终端设备的出现,促进了边缘网络向智能化和虚拟化的方向进行转变,以满足市场的需求。通过网络功能虚拟化(NFV)、软件定义网络(Software Defined Network,SDN)等技术实现软硬件解耦及功能抽象,让网络设备功能不再依赖于专用硬件,使资源的应用变得更加灵活,以应对快速变化的用户需求,而不需要部署和维护物理基础设施[3];且随着智慧交通、智慧城市等智能化边缘应用的实现,愈来愈多的网络业务利用边缘基础设施虚拟化的计算,从而为终端用户提供更加优质的服务。

将用户应用程序转换成为 VNF 实例保证了网络管理的灵活性和便利性,但是随之带来了在运行 VNF 实例的可靠性方面的更多关注。一些特殊应用,例如军事、医疗和化工等方面,对系统的可靠性要求极高,实现这种水平的可靠性是极其困难的。NFV 在 MEC 环境中和专用硬件设备相比,VNF 实例更容易出错和失败[4]。为用户提供可靠的服务,同时满足他们对服务的需求,在传统上保障可靠性的方法主要是利用容错机制,保证每个用户的请求同时备份在多个 VNF 实

基金项目:国家自然科学基金项目(61972414);北京市科技新星项目(Z201100006820082)
This work was supported by the National Natural Science Foundation of China(61972414) and Beijing Nova Program(Z201100006820082).
通信作者:黄霁崴(huangjw@cup.edu.cn)

例上，防止所需的 VNF 实例产生故障。然而，这些基于多重复制的可靠性增强技术需要使用更多冗余的服务器资源，这不可避免地导致了较高的服务成本以及资源消耗。为了进一步降低服务成本，一种有效解决方案是考虑对部分服务进行备份。假设每个用户请求都需要一个具有特定可靠性要求的服务功能链(SFC)进行服务，为了满足用户的可靠性要求，需要保证边缘服务器在处理任务过程中的可靠性，考虑将满足用户需求类型的 VNF 实例放置在不同的服务器上[5]。实际上，大多数用户的服务请求都可以成功完成，因此，我们只需要分析和预测边缘服务器在处理任务过程中可能会产生故障的情况。在 VNF 实例故障之前，准确对其进行备份，能够缓解冗余备份造成的服务提供商成本过高的问题。除此之外，针对边缘服务器的存储空间受限问题，将 VNF 实例放置到其他的边缘服务器上，来满足用户需求。同时，用户请求的迁移需要考虑网络带宽的竞争。

为了进一步解决上述的问题，本文分析了边缘服务系统在处理用户请求的过程中 SFC 的可靠性，并对 VNF 实例进行预测，降低服务的成本；考虑到任务调度过程中的链路竞争，提出了一种启发式的可靠性感知的服务放置算法。本文贡献可以总结如下：

1)在 MEC 环境中制定了一个 VNF 实例的放置问题，其目的是在满足用户可靠性需求的前提下，尽可能降低服务提供商的成本。

2)为了降低过度冗余导致成本过高的问题，采用了基于统一设备架构支持门控循环单元(GRU)神经网络对 VNF 实例进行预测，提前在服务器上放置可能故障的备份 VNF 实例。

3)将 NP-hard 问题转化为凸优化问题，提出了一种可靠性感知的 VNF 实例放置算法，得到服务放置决策。

2 相关工作

为了保护系统不受到攻击，文献[6]提出了一种协调保护机制，该机制同时采用网络层保护和功能级的 VNF 副本。他们专注于最大限度地减少总计算资源成本和网络堵塞。文献[7]提出一种考虑弹性约束的资源分配算法，保护网络服务免受故障影响，利用共享备份网络资源的优势，以降低用于提供弹性分配的资源成本。为了对系统可用性进行评价，文献[8]提出了一种概率方法来衡量系统的实际可用性，并设计出利益相关者可用于在线和离线规划的高效和有效的算法，考虑如何对备份虚拟机进行分配来保证服务器的高可用性。在可靠性优化方面，文献[9]考虑在请求的 VNF 能够并行执行的情况下，优化对 VNF 的部署，目标是在满足请求延迟要求的同时，最大限度地提高利润；并且为了保证服务的可靠性，给每个请求的 VNF 分配一个可以与其他请求共享的备份。文献[10]使用增量方法来确定所需的 VNF 备份数，从而保证服务的可靠性，并提出一种基于共享资源的 VNF 分配策略，权衡给定服务链的所有可靠性、带宽和计算资源消耗。

边缘计算是目前广泛使用的计算范式，在 MEC 环境中考虑部署 VNF 是为了向用户配置网络服务，以降低专用硬件基础设施上的服务成本。Li 等[11]先制定了一个新的 VNF 服务可靠性问题，为了在满足用户可靠性的前提下最大化为用户提供服务的数量，并且必须立即做出准入或拒绝决定，从而最大限度地提高所获得的收入，开发了两种有效的在线算法并证明了算法的有效性。Qu 等[12]提出了一种基于 VNF 分解的备份策略和一种时延感知混合多路径路由方案，以提高 NFV 网络业务的可靠性并降低这些业务体验的时延，且通过仿真实验证明了算法的有效性。文献[13]设计并实现了一个支持 MEC 的 5G 平台，并提出一种在线放置方法增强了 NFV 编排器(NFVO)的智能性，考虑了延迟的嵌入机制。其中 VNF 最初分配给适当的层以及在线调度算法，根据实际流量进行实例化、扩展、迁移和销毁，目标是在不违反应用服务水平协议(SLA)的情况下，最大限度地利用了系统中服务的用户数量。文献[14]在满足用户请求的指定可靠性要求前提下，考虑不同用户具有不同的可靠性需求，并实现网络吞吐量最大化；接着提出了一种新的可靠性感知 VNF 实例放置问题，证明问题是 NP-hard，并设计了常数近似算法进行求解。文献[15]考虑了在 MEC 网络中的移动感知多实例(MAMI)VNF 放置问题，通过用户保持在不同边缘节点覆盖范围内的经验概率，平衡停机时间和资源成本之间的权衡。

虽然针对 VNF 可靠性有一些研究，但对 VNF 故障进行预测并备份来降低计算成本的研究目前还很少，并且考虑到服务器的存储能力有限，合理地进行 VNF 实例的放置和备份具有挑战性，因此我们在进行故障预测的前提下，提出了 RVP 算法，在尽量满足服务请求的前提下，降低服务提供商的成本。

3 系统模型

在边缘计算系统考虑服务过程中的可靠性是最复杂的问题之一，本节首先分析系统在处理任务过程中的延迟和成本，然后分析了服务功能链的故障，并使用门控循环单元神经网络来预测 VNF 实例的故障情况。由于软件故障预测的神经网络训练计算的规模较大，我们提出基于 CUDA 的并行计算并提前对 VNF 实例进行备份。

3.1 边缘服务器场景

网络模型由无向连通图 $G=(V,E)$ 表示[16]，其中 V 和 E 分别表示连接两个节点的物理节点和链路集。不同供应链之间具有传输流量，且供应链中备份不同的 VNF 对服务器流量之间的传输进行约束。定义一组服务器为 $V\in\{1,2,3,\cdots,j\}$，每个服务器的存储容量定义为 $k_j^{\max}$，剩余存储容量表示为 k_j^{re}，不同服务器之间的传输流量表示为 $e_{j,j'}^{f}\in E$，链路之间具有流量的限制，则不同服务器之间的流量限制表示为 $e_{j,j}^{\max}$，且假设 $F\in\{1,2,\cdots,f\}$ 是由网络 G 提供的一组不同类型的 VNF 实例，部署 VNF 实例在边缘服务器 j 所需要的存储资源表示为 k_j^f。本文认为不同用户 $U=\{1,2,\cdots,u\}$ 所提出的服务请求是相互独立的，并将不同用户所提出的服务请求表示为 R_u，且分别对应的任务大小为 D_u。当用户 u 提出一个请求 R_u 时，需要分配请求对应的 SFC，表示为 S_u^f，其中包含一组部署在服务器上不同类型的 VNF 实例进行任务处理。设一个二进制变量 $a_{f,j}\in\{0,1\}$ 表示是否在服务器上 j 部署请求 R_u 所需的 VNF 实例，并且需要满足存储约束，同时需要考虑链路之间的链接。$a_{f,j}=1$ 表示 VNF 实例 f 部署在服务器 j 上，且每个 VNF 实例只放置一次。服务器上部署 VNF 实例的成本为 B^f，否则没部署，部署后的 VNF 可以满足用户的服务。在实际场景中，VNF 实例在处理任务的过程

中可能会发生故障。为了保证服务的可靠性,需要在服务器上重新对该 VNF 实例进行备份来满足用户的请求。考虑到边缘服务器的存储空间有限,不一定有足够的空间满足对该 VNF 实例的备份,因此我们考虑在其他边缘服务器上进行部署,同时还能够避免该服务器可能会产生的物理故障。在服务器进行 VNF 实例备份后,考虑到任务调度过程中需要满足传输流量的约束,如图 1 所示,若用户 u 的任务请求 R_u 需要的 SFC 包括 VNF1 和 VNF3 类型,当服务器 1 上的 VNF1 出现故障的情况下,需要考虑重新在服务器 1、服务器 3 或者服务器 4 上对 VNF 进行部署。但是由于不同的边缘服务器链路传输的流量存在差异(图中利用不同粗细的虚线来表示),以及不同服务器的存储空间有限,因此需要考虑在满足流量约束的前提下,选择合适的服务器进行备份来保证服务的可靠性。

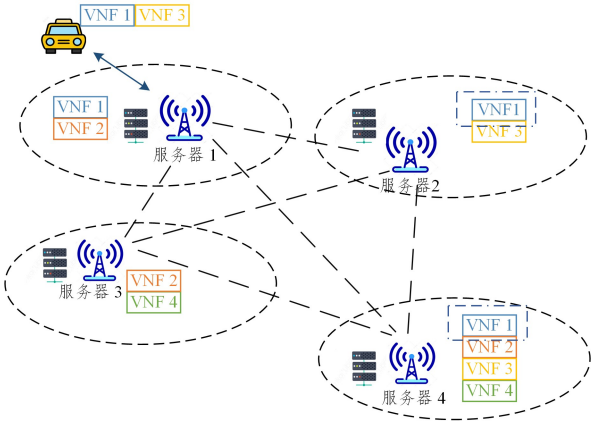

图 1　边缘场景下任务需求放置实例

Fig. 1　Example of task requirement placement in edge computing

在不丧失一般性的情况下,假设一个用户请求 R_u 所需要的 SFC 包括特定类型的一组 VNF,且对应的 VNF 实例 f 所需要的计算资源用 C_u^f 表示。不同的用户任务需要上传到边缘节点进行执行。将传输到边缘节点的过程中产生的时延定义为 T^R:

$$T^R=\frac{D_u}{\eta} \tag{1}$$

其中,η 表示从终端到边缘服务器的链路传输速率,不同 VNF 实例类型 f 的计算能力用 w_n 进行表示,因此得到将用户请求 R_u 在对应的 VNF 实例上进行处理的时间为:

$$T^s=\sum_f\frac{C_u^f}{w_n} \tag{2}$$

其中,处理时间为服务请求所需的计算资源除以不同 VNF 实例类型 f 的计算能力。我们需要考虑 VNF 实例在处理任务的过程中是否会发生故障,引入一个触发器 $I_{f,j}\in\{0,1\}$ 表示 VNF 实例是否可用的预测结果,得到一组不可用的 VNFs 实例,则设一个二进制变量 $b_{f,j}\in\{0,1\}$ 判断将不可用的 VNF 实例备份到哪个服务器上。若备份到当前服务器,则需要考虑部署成本 B^f 和服务计算成本 $B^{j,f}$;若备份到其他服务器上,需要考虑重新部署成本、计算成本和任务传输到重新部署的服务器 j' 上的延迟 $T^c_{j,j'}$,则总的服务延迟需要满足用户的忍耐限度 $T_u^{\max}$。在本文中,若服务器发生故障前未进行备份,则默认超出用户忍耐限度值,则延迟约束表示为:

$$T^R+T^S+\sum_j m_{j,j'}T^c_{j,j'}\leqslant T_u^{\max} \tag{3}$$

其中,二进制变量 $m_{j,j'}\in\{0,1\}$ 中的 j' 是根据 $b_{f,j}\in\{0,1\}$ 得出的服务器连接决策,且链路之间的传输时间是 $T_{j,j'}$ 表示服务器链路之间的连接,当 $j=j'$ 时,链路传输时间为 0。对于服务提供商来说,在尽可能满足用户服务请求下,降低服务成本 B^{Total},主要包括放置和计算成本。考虑到 VNF 实例可能存在故障,因此还需要考虑服务重新配置的成本,因此可以表示为 B^{Total}:

$$B^{\text{Total}}=\sum_j\sum_f(a_{f,j}+I_{f,j}b_{f,j})(B^f+B^{f,j}) \tag{4}$$

3.2　系统模型可靠性分析

本文假设每个请求 R_u 都需要其 VNF 实例 f,其中服务器的故障是相互独立的,并且在满足用户请求的前提下需要保证用户的可靠性需求,因此考虑硬件故障,SFC 的可靠性可计算为:

$$R^S=\prod_1^j a_{f,j}R(j) \tag{5}$$

其中,$R(j)$表示放置的 VNF 实例 f 所在服务器 j 的可靠性,且 $R(j)\in(0,1)$,$R(j)$表示放置 VNF 实例 f 所在的服务器的可靠性,因此在考虑硬件故障下,组成的服务供应链的可靠性表示为 R^s。不同用户在任务请求的过程中,对于可靠性方面有着不同要求。我们针对用户的可靠性需求,将不同用户的可靠性定义为 DR_i,因此在保证用户请求前提下,需满足用户对服务器的可靠性需求。

网络服务供应链(SFC)的可靠性模型:本文没有考虑链路和开关故障,因为现代数据中心通常在任何一对服务器之间有丰富的路径多样性,这可以有效地抵抗这些故障。因此,只考虑 VNF 故障。本文认为 VNF 实例是部署在虚拟机上的。软件执行的可靠性确保在指定的操作条件下提供正确的服务;软件故障是由设计、编码和与用户服务请求相关的固有逻辑问题引起的。本质上,VNF 在处理任务的过程中会产生失败,主要原因是由于系统维护随时间会发生变化。一般来说,系统的工作量越大,软件发生故障的概率就越高。系统工作负载呈现天和时间的二维特征,同时,VNF 实例的软件故障也具有相同的特点。因此,我们采用基于天和时间的二维离散时间序列来表示 VNF 实例的软件故障情况。

本文利用 GRU 神经网络对未来的 VNF 实例可用状况进行预测。该神经网络模型的输入数据根据日期定义为 d 和时间 t[17],并定义 $x^{d,t}=0$ 表示 VNF 实例有故障,不能有效地提供服务;$x^{d,t}=1$ 则表示能够正常提供服务。输入数据定义为 x^t,分别对该神经网络在 t 时刻的重置门 R^t、更新门 U^t、隐藏层 h^t 和输出 H^t 进行定义,并将其表示为:

$$R^t=\sigma(wx^{(rg)}x^t+h^{(d-1,t)}wd^{(rg)}+h^{(d,t-1)}wh^{(rg)}) \tag{6}$$

$$U^t=\sigma(wx^{(ug)}x^t+h^{(d-1,t)}wd^{(ug)}+h^{(d,t-1)}wh^{(ug)}) \tag{7}$$

$$h^t=\tanh(wx^{(og)}x^t+R^t*h^{(d-1,t)}wd^{(og)}+R^th^{(d,t-1)}wh^{(og)}) \tag{8}$$

$$H^t=(1-U^t)*h^t+U^th^{(d,t-1)}+U^th^{(d-1,t)} \tag{9}$$

其中,$\sigma\in(0,1)$和 $\tanh\in(-1,1)$分别为激活函数;wx,wd 和 wh 分别表示权重。故障预测的 GRU 神经网络都与一系列训练数据相关联。因此,训练需要大量的 GRU 所预测模型的时间。根据这些观察结果,为每个 GPU 线程分配了一个隐层节点参数的计算值,所有节点都可以在一个 GPU 块中并行计算。采用 GPU 对 GRU 神经网络模型进行并行训练是一种有效的方法,因此,GRU 神经网络模型训练采用支持 CUDA 的块和线程级并行计算。如果没有发生故障,则服务

链的所有 VNF 实例都可以直接进行处理任务。当我们通过 GRU 进行预测，判断 VNF 将要发生故障时，需要提前备份，将需要的 VNF 类型提前部署到当前服务器或者其他的服务器上，在保证用户可靠性的前提下，同时需要考虑服务器之间的链路传输，满足用户忍耐度约束限制。

4 问题描述

4.1 问题建模

本文的目的是在满足用户延迟忍耐和链路传输下，最小化服务提供商的成本。总的成本可以表示为：

$$\min B^{\text{total}} \tag{10}$$

$$\text{s.t. } T^{R}+T^{S}+\sum_{j} m_{j,j'} T^{c}_{j,j'} \leqslant T_u^{\max}, \forall f\in F, \forall j\in J \tag{C1}$$

$$\sum_{j} e^{f}_{j,j'} \leqslant e^{\max}_{j,j'}, \forall f\in F, \forall j\in J \tag{C2}$$

$$\sum_{f} a_{f,j} k_{f,j} + b_{f,j} k_{f,j} \leqslant k_j^{\max}, \forall f\in F, \forall j\in J \tag{C3}$$

$$\prod_{j} R(j) \geqslant DR_u, \forall j\in \{j \mid a_{f,j}=1\} \tag{C4}$$

$$\sum_{j} a_{f,j} \leqslant 1, \forall f\in F, \forall j\in J \tag{C5}$$

$$\sum_{j} b_{f,j} \leqslant 1, \forall f\in F, \forall j\in J \tag{C6}$$

$$a_{f,j}, I_{f,j}, b_{f,j}, m_{j,j'} \in \{0,1\}, \forall f\in F, \forall j\in J \tag{C7}$$

其中，(C1)表示用户对时间的忍耐约束；(C2)表示对链路传输的约束；(C3)表示服务器的存储能力受限制；约束(C4)是用户对服务器可靠性的要求；约束(C5)是对 VNF 实例的放置决策；(C6)是对 VNF 实例的备份决策；约束(C7)表示 VNF 实例放置、备份以及可用性，和服务器之间连接决策的取值范围，将优化问题视为 NP-hard 问题[18]。解决这类问题具有挑战性，本文考虑将问题进行转化为线性化问题，并进行松弛，引入一个新的变量 $y_{f,j}=I_{f,j}b_{f,j}$ 进行建模[19]，则需要考虑新的约束条件为：

$$\begin{cases} y_{f,j} \leqslant I_{f,j} \\ y_{f,j} \leqslant b_{f,j} \\ y_{f,j} \geqslant b_{f,j} + I_{f,j} - 1 \end{cases} \tag{11}$$

前两个不等式对 $y_{f,j}$ 的下界进行约束，保证当 $I_{f,j}$ 和 $b_{f,j}$ 的值均为 0 时，$y_{f,j}$ 的值为 0；如果当 $I_{f,j}$ 和 $b_{f,j}$ 的值均为 1 时，第三个不等式约束保证了辅助变量的值为 1，因此约束保证了辅助变量的取值范围 $y_{f,j}\in\{0,1\}$。为了将目标问题转化为凸优化问题，将二进制变量松弛为连续变量，则变量 $I_{f,j}$，$b_{f,j}$，$y_{f,j}$，$m_{j,j'}\in[0,1]$，因此松弛后的问题转化为凸问题。将松弛后的二进制变量看成连续变量，由于决策是不可以拆分的，因此可以选择若干值中的最大值作为最后的决策结果，将其取值为整数 1，对应的是 VNF 实例的放置和备份决策。

4.2 算法描述

通过上述的分析和说明，提出了可靠性感知的 VNF 实例放置(RVP)算法，将详细描述在考虑 VNF 实例故障的情况下对 VNF 实例进行放置和备份，对 VNF 实例提前进行预测，提前对可能故障的 VNF 实例进行备份，降低冗余备份的成本，并且考虑链路传输和用户忍耐度，得到最优的放置和备份决策，最小化服务提供商的成本。

对于 RVP 算法的具体实现步骤如算法 1 所示。首先随机放置一组用户请求可能需要的 VNFs 实例，并且考虑到服务器的存储容量限制，对可能故障的 VNF 实例进行预测，得到预测结果(第 1—2 行)。计算放置决策是否满足约束条件(第 3—8 行)，对可能故障的 VNF 实例进行备份，得出最小化成本的方案，更新服务器资源和用户请求数，直到循环结束(第 9—17 行)。

算法 1 RVP 算法

输入：用户请求 R_u；

输出：$a_{f,j}$，$b_{f,j}$ 和 B^{total}

1. 初始化：随机在边缘服务器上部署一批用户所需虚拟 VNF 实例，此时服务器的存储容量为 k_j^{re}；
2. 使用 CUDA 的并行 GRU 训练，预测虚拟机是否故障 $I_{f,j}$；
3. while 在时间段 t 内用户产生一组任务请求 do
4. for 对于每个任务 R_u 所需的 SFC 类型 do
   if 预测的 VNF 实例未发生故障 then
   完成 S_u^f 所需的处理时间；
   完成 S_u^f 所需的成本 B；
5. if 不满足用户忍耐限度 $T_u^{\max}$ 和传输链路限制 then
6. 拒绝用户任务；
7. else
   找到成本最小的 VNFs 实例放置决策；
8. end
9. else
10. for 服务器 j 对故障 VNFs 实例进行部署 do
11. 重新部署完成 S_u^f 所需的处理时间；
12. 重新部署完成 S_u^f 所需的成本 B；
13. end
14. 重复步骤 5—7；
15. end
16. 更新服务器剩余存储资源和用户请求数量 R_u.
17. end

5 实验验证

5.1 实验设置

在边缘服务器上处理用户请求，考虑到任务的大小分布在{0,10} MB 范围内，每个 VNF 对应的计算能力在{0.1,…,0.5} GHz 范围内。对于存储来说，边缘服务器的存储空间对应在{8,16}GB 范围，放置对应的 VNF 实例所需要的存储空间在{0.1,2}GB 范围内。为了训练 LSTM 模型，本文使用了北京 301 医院停车场两年的服务日志，通过分析数据，得到服务异常和对应的时间，将该数据作为训练数据进行故障预测。

如图 2 所示，预测每分钟的服务器的可用性，并且每 5 min 记录一次，得到可靠性预测值和实际值是否一致。通过实验可知，GRU 预测的准确率要高于其他两种算法。

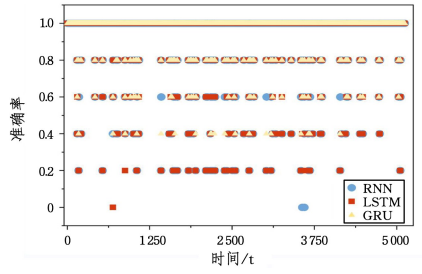

图 2 预测准确度对比

Fig. 2 Comparison of prediction accuracy

5.2 对比算法

在本节中，通过仿真演示了所提出算法的有效性，故本文将所提出的算法和其他两种算法算法进行对比，证明过度冗余或者未及时备份会对服务成本和服务质量产生影响。其

中，贪婪算法是指对每个 VNF 实例进行备份，不考虑过度备份对成本的影响；随机放置是指对可能使用到的 VNF 实例随机进行备份。在本文中，如果 VNF 实例故障但是没有提前进行备份，则该请求被拒绝。

如图 3 所示，本文分别通过改变用户数量和边缘服务器的数量对所提出的算法以及其他两种算法进行对比。随着所需 VNF 实例数量的增加，成本是呈增长的，并且提出的 RVP 算法要明显优于其他两种算法。

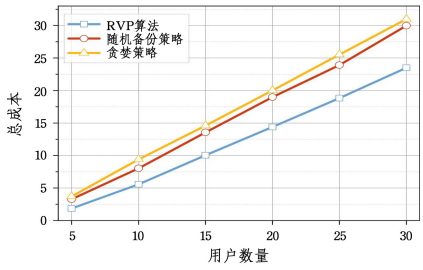

图 3　成本与用户数量之间的关系

Fig. 3　Relationship between cost and the number of users

当服务器数量不变时，如图 4 所示，随着用户数量的增加，服务拒绝率也在增长，并且贪婪策略由于边缘服务器的容量受限，会影响服务成功率，但是优于随机备份策略。因此通过实验结果看出，本文所提出的 RVP 算法的服务拒绝率最低，且随着用户数量的增加，优势越明显。

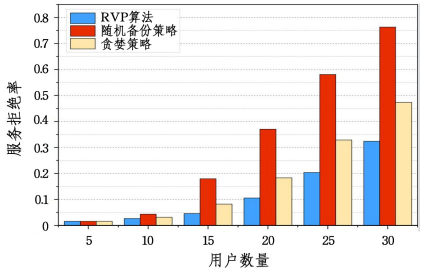

图 4　服务拒绝率与用户数量之间的关系

Fig. 4　Relationship between service rejection rate and the number of users

如图 5 所示，随着服务器数量的增加，用户数量保持一定，则服务拒绝率会随之降低，因为能够满足服务放置的选择随着边缘服务器数量增加而增多，但当服务器数量增加到一定程度时拒绝率变化幅度就会变小，因此合理分配服务器数量是必要的。

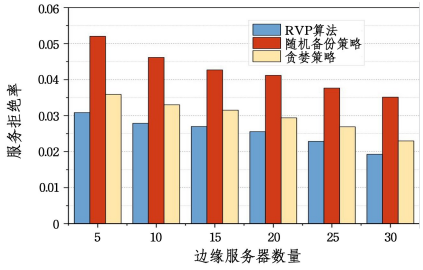

图 5　服务拒绝率与服务器数量之间的关系

Fig. 5　Relationship between service rejection rate and the number of edge servers

结束语　本文研究了在边缘服务器的存储受限的前提下，考虑 VNF 实例失效和服务器故障对服务质量的影响。为了保证服务质量，降低冗余备份导致的成本过高问题，提出了 RVP 算法，对 VNF 实例进行预测，放置用户服务所需要的 VNF 实例并对可能故障的实例进行备份，得到成本最小化的放置决策，并通过实验证明提出方法的有效性。本文只针对用户请求在边缘服务器上进行计算的场景分析，只能解决较小的服务请求，针对大型的应用场景需要考虑本地-边缘-云的协同计算；其次是没有考虑到任务之间的依赖关系。接下来将研究一个基于本地-边缘-云的三层协作的服务框架，考虑任务之间的依赖关系进行分析。

参 考 文 献

[1] PALADE A，KAZMI A，CLARKE S，et al. An evaluation of open source serverless computing frameworks support at the edge[C]// IEEE World Congress on Services. SERVICES，2019：206-211.

[2] LIN L，YANG S，MIN Z，et al. Effective replica management for improving reliability and availability in edge-cloud computing environment [J]. Parallel and Distributed Computing，2020，143：107-128.

[3] ZHANG J，ZENG D，GU L，et al. Joint optimization of virtual function migration and rule update in software defined NFV networks[C]// Global Communications Conference. IEEE GLOBECOM，2017：1-5.

[4] FAN J，GUAN C，ZHAN Y，et al. Availability-aware mapping of service function chains[C]// Conference on Computer Communications. IEEE INFOCOM，2017：1-9.

[5] WANG Y，SHU Z，ZHONG Y，et al. Service function chain placement algorithm based on VNF instance sharing [J]. Application Research of Computers，2023：1-8.

[6] KONG J，KIM I，WANG X，et al. Guaranteed-Availability Network Function Virtualization with Network Protection and VNF Replication [C] // Global Communications Conference. IEEE GLOBECOM，2017：1-6.

[7] BECK M，BOTERO J，SAMELIN K，et al. Resilient allocation of service Function chains[C]// Network Function Virtualization and Software Defined Networks. IEEE NFV-SDN，2016：128-133.

[8] CASAZZA M，FOUILHOUX P，BOUET M，et al. Securing virtual network function placement with high availability guarantees[C]// IFIP Networking Conference and Workshops. IFIP Networking，2017：1-9.

[9] WU Y，ZHENG W，ZHANG Y，et al. Reliability-Aware VNF Placement Using a Probability-Based Approach [J]. IEEE Transactions on Network and Service Management，2021，18(3)：2478-2491.

[10] QU L，KHABBAZ M，ASSI C. Reliability-Aware Service Chaining In Carrier-Grade Softwarized Network [J]. IEEE Journal on Selected Areas in Communications，2018，36(3)：558-573.

[11] LI J，LIANG W，HUANGM，et al. Reliability-Aware Network Service Provisioning in Mobile Edge-Cloud Networks [J]. IEEE Transactions on Parallel and Distributed Systems，2020，31(7)：1545-1558.

[12] QU L，ASSI C，KHABBAZ M，et al. Reliability-Aware Service Function Chaining With Function Decomposition and Multipath Routing [J]. IEEE Transactions on Network and Service Mana-

gement, 2020, 17(2): 835-848.

[13] ALAHMAD Y, AGARWAL A. VNF Placement Strategy for Availability and Reliability of Network Services in NFV[C]// International Conference on Software Defined Systems. IEEE SDS, 2019: 284-289.

[14] SARRIGIANNIS I, RAMANTAS K, KARTSAKLI E, et al. Online VNF Lifecycle Management in an MEC-Enabled 5G IoT Architecture [J]. IEEE Internet of Things Journal, 2020, 7(5): 4183-4194.

[15] WEI Q, HAN P, LIU Y, et al. Mobility-Aware Multi-Instance VNF Placement in Mobile Edge Computing Networks[C]// International Wireless Communications and Mobile Computing. IEEE IWCMC, 2021: 1303-1308.

[16] HUANG M, LIANG W, SHEN X, et al. Reliability-Aware Virtualized Network Function Services Provisioning in Mobile Edge Computing [J]. IEEE Transactions on Mobile Computing, 2020, 19(11): 2699-2713.

[17] TANG X, LIU Y, ZENG Z, et al. Service Cost Effective and Reliability Aware Job Scheduling Algorithm on Cloud Computing Systems [J]. IEEE Transactions on Cloud Computing, 2023, 11(2): 1461-1473.

[18] XING H, LIU L, XU J, et al. Joint task assignment and resource allocation for d2d-enabled mobile-edge computing [J]. IEEE Transactions on Communications, 2019, 67(6): 4193-4207.

[19] MALLACH S. Compact linearization for binary quadratic problems subject to assignment constraints [J]. 4OR-Q J Oper Res, 2018, 16: 295-309.

LIANG Jingyu, born in 1998, Ph. D student, is a member of CCF (No. C6280G). Her main research interests include reliability and edge computing.

HUANG Jiwei, born in 1987, professor, Ph.D supervisor, is a senior member of CCF (No. 20352S). His main research interests include services computing, Internet of Things and edge computing.