

可重构柔性网络资源管理与服务映射方法研究

王保进, 汪斌强, 王志明

(国家数字交换系统工程技术研究中心, 河南 郑州 450002)

摘要: 提出了基于资源重组的可重构柔性网络(RFNet)。RFNet中的网络服务是构件的有机组合,并且能够根据新的业务需求,利用构件复用思想,从软硬件出发重构出满足需求的新的网络服务。首先介绍RFNet的体系结构,然后提出RFNet的“资源管理—承载网映射—网络服务映射”的流程机制。经过评估,提出的RFNet“资源管理—承载网映射—网络服务映射”机制能够很好地支持差异化服务的构建与共存,并且基于负载均衡的可重构服务承载网映射算法RSCNM在网络构建成功率、最大节点强度、平均链路利用率和构建平均收益上具有明显的优势。

关键词: 可重构柔性网络; 资源管理; 资源分配; 网络服务映射; 可重构路由交换平台; 构件

中图分类号: TP393

文献标识码: A

文章编号: 1000-436X(2012)08-0021-14

Research on resource management and service mapping method for reconfigurable flexible network

WANG Bao-jin, WANG Bin-qiang, WANG Zhi-ming

(National Digital Switching System Engineering & Technology Research Center, Zhengzhou 450002, China)

Abstract: Reconfigurable flexible network (RFNet) based on resource re-composition was proposed. The network service in RFNet was composed of special components, and could construct the new network service to satisfy a coming traffic demand by component composition technology in both software and hardware area. After introducing the architecture of RFNet, a process flow mechanism of resource management, carrying network mapping and network service mapping was proposed. By simulation evaluation, the mechanism was confirmed to provide the support of heterogeneous networks construction and coexistence, and especially the reconfigurable service carrying network mapping (RSCNM) algorithm based on load balance had an advantage over network successful construction ratio (SCR), maximum node stress, average link utilization rate and average revenue.

Key words: reconfigurable flexible network; resource management; resource allocation; network service mapping; reconfigurable router switch platform; component

1 引言

随着互联网业务类型的不断丰富,网络规模的不断扩大,网络技术必须不断地进行革新,才能适应这些变化。传统互联网依靠设备升级、带宽扩展以及增加新的协议算法等方式来满足业务需求的

变化,导致互联网体系结构日趋复杂。根本原因在于当前互联网体系结构是僵化的^[1],只能依靠升级与扩展来适应新业务。

目前,国内外都制定了具有前瞻性的新型互联网研究建设计划,核心思想都是探索一种能够支持多业务融合的一体化网络架构,以解决网络用户业

收稿日期: 2011-12-05; 修回日期: 2012-05-20

基金项目: 国家重点基础研究发展计划(“973”计划)基金资助项目(2012CB315901, 2012CB315905)

Foundation Item: The National Basic Research Program of China (973 Program)(2012CB315901, 2012CB315905)

务多样性等急迫需求。国外对新一代互联网体系结构的研究计划主要有美国的 GENI^[2]和 FIND^[3], 欧盟的 FIRE^[4]和 4WARD^[5], 以及日本的 AKARI^[6]等。2005 年 8 月, NSF(national science foundation)宣布了全球网络创新环境(GENI, global environmem for network innovations)计划, GENI 基于 PlanetLab 已建成的实验床平台, 其核心目标是改变现有网络和分布式系统设计的方式; 未来因特网设计(FIND, future internet design)与 GENI 存在重要联系, 其研究方向基本被 GENI 涵盖在内, 只是其更侧重未来互联网体系框架的研究; 作为欧盟 FP7^[7]在 ICT 领域的重要组成部分, 未来因特网研究实验(FIRE, future internet research and experimentation)是欧盟为应对未来互联网研究、应用面临的诸多挑战而实施的大型研究计划, 其最终目标是通过逐步将现有和未来互联网技术实验床平台联合, 建设一个动态、可持续、大规模的欧洲实验基础设施平台, 从而为欧盟互联网技术发展提供多学科综合的研究实验环境; 4WARD 是 FP7 在网络技术领域的代表性项目, 一方面通过创新, 改造现有单个网络架构, 克服不足, 提高其性能, 另一方面研究一种总体网络框架, 用于实现各种不同网络架构的协作运行, 从而走出目前网络僵化的窘境; AKARI 是日本 NICT(national institute of information and communications technology)2006 年启动的新一代网络研究项目, 其目标是在 2015 年前研究出一个全新的网络构架, 并完成基于此网络架构的新一代网络的设计。

国内具有代表性的研究主要有面向服务提供的可重构柔性网络^[8] (RFNet, reconfigurable flexible network), 该网络以可重构路由交换平台^[9]为支撑, 通过构建多个可重构服务承载网 (RSCN, reconfigurable service carrying networks)的方式实现多种异构网络并存, 从而提供一种解决网络僵化问题且兼容现有网络的创新思路。

可重构服务承载网的资源管理、分配和部署技术是柔性网络理论体系的关键内容, 目前国内外相关研究主要围绕虚拟网^[10]映射 (又称赋值、资源分配) 算法展开。虚拟网映射算法大致分为 2 类^[11]: 静态方式和动态方式。前者在虚拟网的活动周期中对映射方案不做任何调整; 后者能根据物理网络和虚拟网的当前状态对映射方案进行适应性调整。对于静态方式, Ricci 等^[12]最早提出了 Assign 算

法求解实验床中的子网资源分配问题; 随后的研究大多将虚拟网映射问题分解为节点映射和链路映射 2 个步骤, 并分别提出各种启发式算法和线性规划模型^[13~15]; 最近的研究正逐步试图将节点映射和链路映射关联起来, 比如 Chowdhury 等^[16]提出的 D-ViNE 和 R-ViNE 算法, 以及 Jens 等^[17]的子图同构检测算法 vnmFlib。对于动态方式, He 等^[18]提出了依据虚拟网性能动态调整链路带宽分配的 DaVinci 算法; Rahman 等^[19]针对物理拓扑故障提出了一种保证虚拟网生存性的映射算法; Cai 等^[20]则针对物理拓扑变化的一般情况, 提出一种基于代价最小原则的节点迁移与重映射算法。

上述研究的不足之处在于: 大多数研究只限于发现新的虚拟网映射模型, 缺乏一种有效支持大规模网络的资源管理机制; 在对虚拟网资源进行分配后, 缺乏一种通用的网络资源管理与部署机制, 使得相关研究失去实际意义。针对这些不足, 本文给出一种有效的可重构柔性网络资源管理方法, 并在此基础上提出可重构服务承载网资源分配方法和网络服务部署方法, 最后针对可重构柔性网络特性, 实现了一种标准化的网络资源管理与网络服务部署机制。

2 可重构柔性网络体系

2.1 关键概念

面向服务提供是 RFNet 的核心思想。由于网络业务多样性、差异化的趋势, 采用不变的网络体系结构或服务模式支撑所有业务需求, 即“one size fits all”在技术上是极难实现的, 设计复杂度和设备成本也会很高; 从用户体验质量 (QoE) 角度来说, 针对不同业务匹配不同的网络体系和服务模式, 即“one size fits one”是最理想的, 但如此细粒度地对业务进行适应会带来管理复杂度和管理成本高等问题。RFNet 综合上述考虑, 采用中等粒度的业务适应方式, 针对不同业务类型提供不同的网络服务 (见本文定义 1), 即“one size fits group”模式。该模式既保持了对业务多样性的适应, 又能尽量减少网络服务的更换频率。

定义 1 网络服务 (RFNet service) 是指服务提供者为满足一类网络业务需求而提供的一套运行在可重构路由交换平台之上的服务单元 (见定义 2) 集合。在可重构路由交换平台的支持下, 网络服务可被服务提供商部署、重构和释放。

定义 2 服务单元 (service unit) 是网络服务的基本组成部分, 每一类业务需求都可以分解成若干功能的有机组合, 本文把实现此种功能且具有重构特性的实体称之为 RFNet 下的服务单元。

定义 3 在 RFNet 中, 重构 (reconfiguration) 是指通过构件级别的变更完成不同网络服务之间的转换, 以及同种网络服务不同配置之间的转换。

定义 4 构件 (component) 是服务单元的基本组成部分, 是具有一定网络处理能力的实体及其相关描述。在 RFNet 中, 构件可以被明确识别、复用和重构, 分为软件构件和硬件构件 2 大类。构件是可重构柔性网络中的重要资源。

定义 5 可重构服务承载网是针对用户业务需求而抽象出来的逻辑网络, 该网络在结构上能够映射成为实际物理网络的子图, 在资源上是实际物理平台资源的一个分片 (slice)。RFNet 中, 可重构服务承载网支持构建、重构、迁移和注销等行为。

2.2 分层模型

RFNet 采用 4 层结构的分层参考模型, 如图 1 所示, 自顶向下 4 层结构分别是: 应用层、业务层、服务层和资源层, 各层所代表的含义如下。

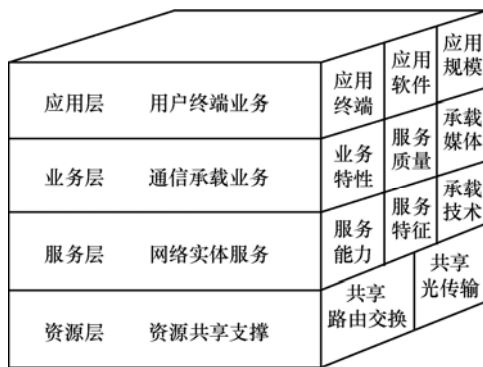


图 1 可重构柔性网络分层参考模型

应用层: 可重构柔性网络的应用层不同于传统网络应用层定义, 它包含了现有及未来可能出现的各种用户应用, 可用来描述用户终端业务。通过对目前用户应用需求的分析, 定义用户业务的产生均可以使用应用终端、应用软件和应用程序规模来确定, 应用终端包括话机、移动终端、计算机等, 应用程序指的是终端中具体执行人机交互的应用程序, 应用程序规模指明了业务规模的大小。上述应用层定义从用户业务生成源出发, 为下层的业务层提取业务特征确定服务质量参数提供了便利。

业务层: 业务层主要完成通信承载业务, 提

取应用层生成业务的特征并且将用户业务聚类, 按照业务特性、服务质量和承载媒体分别确定相应的参数。该层完成的功能包括媒体网关功能, 信令网关功能, 边界路由节点执行的分类、整形、标记等接入控制功能。该层描述了用户业务的特征, 为可重构柔性网络构建服务承载网提供参考标准。

服务层: 该层是可重构柔性网络的核心层, 基于资源共享层提供的物理资源, 通过构建可重构服务承载网的形式为业务层提供所需的网络服务。按照服务的属性和特征, 根据松散耦合和简化的原则, 把网络服务分为服务能力、服务特征和承载技术 3 种属性。根据上层业务层提供的参数分别确定相应的网络服务能力和服务特征, 选择合适的承载技术, 与传统网络服务相比, 可重构柔性网络可以根据用户需求构建网络, 按照业务特性提供服务能力, 从业务需求出发保证服务质量。

资源层: 资源层是一种资源可共享、节点可重构, 能够提供底层网络资源的物理网络, 资源共享层由可重构路由节点、光传输设备等组成, 为可重构柔性网络提供共享的物理网络资源和光传输资源。

2.3 组网架构

RFNet 的组网架构如图 2 所示, 包括 6 种元素: 可重构管理平台 (RFNet controller, 以下简称管理平台)、构件库系统、可重构路由交换平台、可重构光网络设备、媒体网关以及用户终端。各元素的特点如下。

1) 可重构管理平台。该平台负责感知和管理全网资源: 发现可重构路由交换平台的服务能力并对其进行组合以支撑相应的用户业务; 将用户的业务特性参数转换为 RSCN 构建请求, 并根据当前网络资源状况及可重构路由交换平台服务能力构建 RSCN; 根据构建指令, 从构件库中下载相关构件并加载运行从而重构出新的节点服务能力, 支撑新的网络服务。可重构管理平台在 RFNet 中扮演了服务提供商和控制代理的角色。

2) 构件库系统。该系统存储有包括第三方厂商开发的标准构件。构件存储的形式为源代码、二进制码以及相关描述文档和配置脚本等多种物理形态, 文档遵从一定的格式标准。可重构路由交换平台根据管理平台的指令, 通过从构件库中下载相关

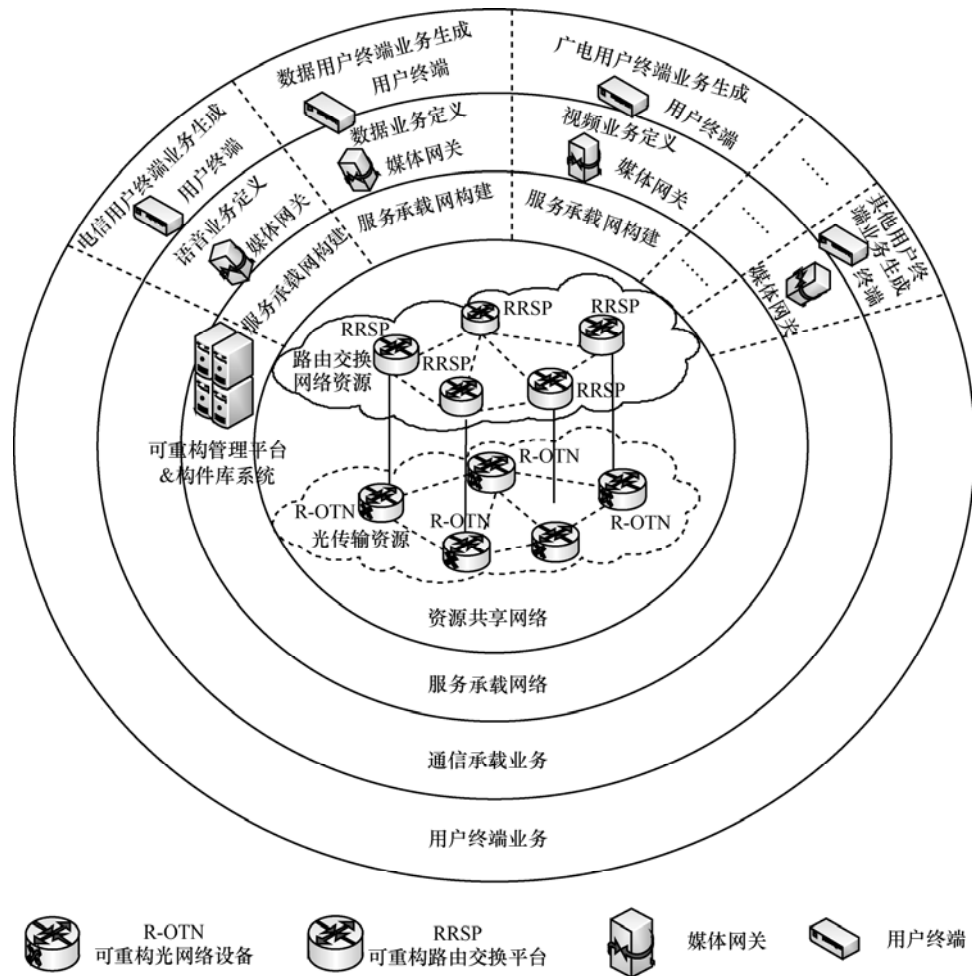


图 2 可重构柔性网络组网架构

构件并加载运行从而重构出新的网络服务，提供新的业务承载能力。构件库系统在 RFNet 中扮演了构件提供商的角色。

3) 可重构路由交换平台。该平台为各种网络服务的运行提供链路带宽、交换、缓存、CPU、FPGA 器件等物理资源；提供构件运行及相关功能处理、生成网络服务的支撑环境；提供资源代理 (resource agent) 和重构代理 (reconfiguration agent) 等管理模块，以支持资源管理和设备级重构。可重构路由交换平台在 RFNet 中扮演了设施提供商的角色。

4) 可重构光网络设备。可重构路由交换平台既可以直接互联又可以通过可重构光网络设备进行互联。利用可重构光网络设备的重构特性可以方便地实现可重构柔性网络的拓扑可变。

5) 媒体网关。媒体网关对不同的应用需求编码

进行聚类，并将对应的业务特性、服务质量和承载媒体参数发送至可重构管理平台。

6) 用户终端。用户终端产生应用需求，并将应用需求参数进行编码发送至媒体网关。应用需求参数包含终端类型、应用软件和应规模等重要信息。

3 可重构柔性网络的资源管理机制

为了提高网络的可扩展性，将 RFNet 的资源管理分层和分域，通过构建较小的资源管理域，能有效地控制由于网络规模扩大带来的复杂度和相应开销的增加。RFNet 的资源管理主要包括预置信令网 (PSN, pre-existed signaling network) 的构建、资源感知和资源分配。

3.1 预置信令网

预置信令网是一种特殊的可重构服务承载网，

在 RFNet 初始化时就被部署到所有节点上，主要用于传递资源代理层内部以及资源代理层与资源管理层之间的交互信令，为构建其他可重构服务承载网提供管理环境支持。预置信令网从功能角度分为资源代理层 (resource agent layer) 和资源管理层 (resource manage layer)，从地域上又分为各个资源管理域，其整体结构如图 3 所示。

资源代理层由可重构路由交换平台组成，可重构路由交换平台原则上是可以被多个服务提供商所管辖的，所以不应再根据服务提供商进行分域。在 RFNet 中，同一服务提供商的路由交换平台是根据网络所处地理位置及其规模进行分域的，不同服务提供商所辖的路由交换平台可以存在重叠。可重构路由交换平台的主控设计中有资源代理模块，该模块负责感知路由器节点的剩余资源情况，以及邻接链路的状态信息等，并周期性地将其反馈给资源管理层。

资源管理层由可重构柔性网络管理平台组成，每个管理平台负责指定域的资源管理，包括：接受用户的可重构服务承载网构建请求，对域内资源进行感知，构建满足用户业务需求的可重构服务承载网，管理不同可重构服务承载网的资源分配使用情况等。资源管理层之间还存在域间资源管理逻辑，包括域间资源感知和域间资源分配。

预置信令网的构建可分为域内和域间 2 个范围。域内构建是指管理平台与其所管控的可重构路由交换平台间构建一个可重构服务承载网，采用的是最小生成树算法，即以管理平台为根，可重构路由交换平台为中间节点和叶子节点，最小化管理平台到路由节点间的传输延时与网络资源开销 (如图 3 中所示的虚线部分)。域间构建是指为管理平台间传递信令而构建的可重构服务承载网，管理平台间

的拓扑连接关系可以是全连接 (如图 3 中所示的加粗虚线部分)、环状或者共享树结构，不同的拓扑连接关系在网络资源开销、信令传递的实时性、可靠性上会有所不同，如果采用环状连接，则路径规划将转化为旅行商问题 (TSP, travel salesman problem)。

3.2 资源感知

管理平台在进行可重构服务承载网构建时，需要根据重构网策略进行规划，只有充分掌握可重构路由交换平台的资源使用情况，才能进行最优的规划。由于可重构服务承载网构建请求可以是跨域的，所以 RFNet 的资源感知分为：域内资源感知和域间资源感知。

RFNet 的资源感知是通过域内网元控制 (NEC, network element control) 协议和域间状态公告 (DSA, inter-domain status announcement) 协议进行的。这 2 种协议都是基于 web service 通信，在域内资源感知时，管理平台与可重构路由交换平台之间通过 NEC 协议进行通信，从而使管理平台感知域内路由节点的拓扑连接情况、邻接链路情况、节点性能、节点功能以及节点状态等信息；在域间资源感知时，管理平台之间通过 DSA 协议维护域间状态的同步。例如，当进行资源感知时，管理平台通过 NEC 协议将资源感知请求发送到所辖域的可重构路由交换平台，可重构路由交换平台将该节点的拓扑结构、邻接链路状态、节点性能、功能以及状态返回给管理平台。当涉及到跨域构网时，管理平台还需要将所辖域内的整体资源信息通过 DSA 协议通告给其他所有管理平台。通常情况下，管理平台周期性地感知资源，但当节点状态 (如邻接状态) 发生变化时，节点通过 NEC 协议将新的资源信息发送给管理平台，从而缩短资源感知的

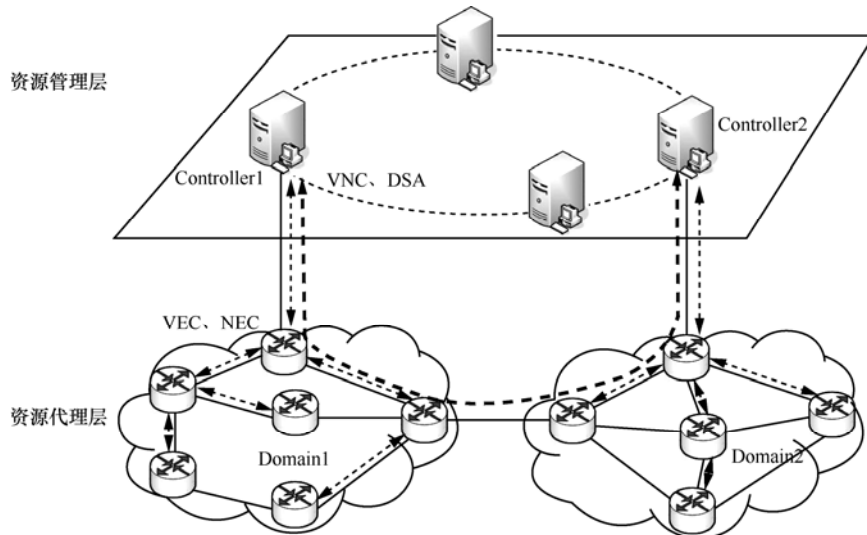


图 3 RFNet 的资源管理框架

更新时间。

3.3 资源分配

可重构服务承载网的构建过程实际上就是底层可重构路由交换平台的资源分配过程，管理平台在完成对资源的感知之后，就可以根据构网策略进行资源的最优分配。可重构服务承载网的构建请求可以是跨域的，所以 RFNet 的资源分配分为：域内资源分配和域间资源分配。

RFNet 的资源分配是通过域内可重构服务承载网元控制 (VEC, intra-domain virtual element control) 协议和域间可重构服务承载网控制 (VNC, inter-domain virtual network control) 协议进行的。与资源感知协议类似，资源分配协议也是基于 web service 通信的。当管理平台进行可重构服务承载网构建时，管理平台根据资源感知情况以及构网策略作出规划，如果涉及到跨域构网，管理平台将所需的其余域的构网需求通过 VNC 协议发送到所对应的管理平台中，这些管理平台再规划出本域内的构网方案。这样，无论域内或域间请求，管理平台只需将构网指令通过 VEC 协议发送至相关可重构路由交换平台的重构代理模块中，并在收到其重构成功响应后完成可重构服务承载网的构建。

4 可重构服务承载网映射算法与机制

映射是构建可重构服务承载网的重要过程，是可重构柔性网络管理平台的重要功能。所谓可重构服务承载网映射，即根据用户的构建需求及物理网络当前的资源状况和服务能力，通过映射算法在构建需求与网络资源之间进行匹配，以获得最优的网络资源分配方案进行网络服务的部署，生成可重构服务承载网。

4.1 映射模型

一个简单的可重构服务承载网映射实例如图 4 所示，该可重构服务承载网请求有 3 个虚拟节点 a、b、c 以及虚拟链接 ab、ac 组成，物理网络由 A、B、C、D、E、F 节点及其相应物理链接组成。经过映射，得到满足节点约束和链路约束的最优匹配结果，即虚拟链路(a,b)对应物理路径(A,B,C,D)，虚拟链路(a,c)对应物理路径(A,B,F)。

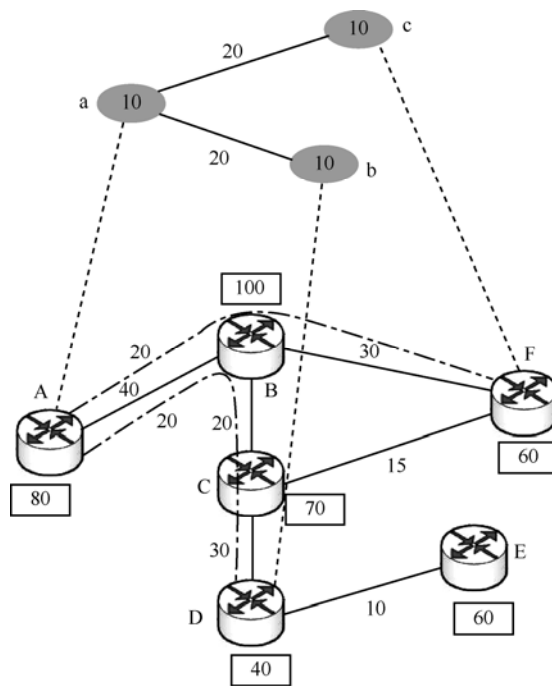


图 4 可重构服务承载网映射实例

上述实例反映出可重构服务承载网映射本质上是一个关于图的最优问题，所以本文将可重构服务承载网映射抽象成一个图模型，并在此基础上根据目标函数求最优解。可重构服务承载网映射模型的建立如下。

底层物理网络模型。将物理承载网络用无向图 $G^S = (N^S, L^S, A_N^S, A_L^S)$ 表示，其中， N^S 和 L^S 代表物理网络的节点集合和边集合， A_N^S 和 A_L^S 代表物理网络的节点属性和链路属性，如节点交换能力、CPU 个数、链路带宽、延迟等。此外，在底层物理网络模型中，所有的无环路径集合 P^S 也需要被定义。

可重构服务承载网请求模型。将可重构服务承载网构建请求用无向图 $G^V = (N^V, L^V, C_N^V, C_L^V)$ 表示，其中 N^V 和 L^V 代表可重构服务承载网的虚节点集合和虚边集合， C_N^V 和 C_L^V 代表可重构服务承载网请求对虚节点和虚链路的约束条件，比如对虚节点的 CPU 资源、地理位置等的约束，以及对虚链路的带宽、延迟等的约束。

可重构服务承载网映射模型。可重构服务承载网映射问题可以描述为从 G^V 到 G^S 子集的一个满足 G^V 中约束条件的映射行为，表示形式如下：

$$M : G^V \mapsto (N', P', R_N, R_L)$$

其中， $N' \subseteq N^S$ ， $P' \subseteq P^S$ ， R_N 和 R_L 分别表示为可重构服务承载网请求分配的节点资源和链路资源。从

上述映射形式可以看出可重构服务承载网映射过程其实可以分为 2 个方面：节点映射和链路映射。具体表示形式如下。

节点映射：

$$M^N : (N^v, C_N^v) \mapsto (N', R_N)$$

链路映射：

$$M^L : (L^v, C_L^v) \mapsto (P', R_L)$$

目标函数。为了使底层物理网络能够承载更多的可重构服务承载网映射请求，本文基于负载均衡策略制定目标函数。首先，定义 t 时刻底层物理网络 G^S 的各节点负载 $N_{\text{load}}(t, u)$ 和链路负载 $L_{\text{load}}(t, e)$ ，其中 $u \in N^S$ ， $e \in L^S$ 。可重构服务承载网映射模型对节点和链路的资源属性进行了抽象化描述，本文将着重关注其中的链路带宽资源和节点 CPU 计算资源，进而得到对节点负载和链路负载的具体定义。

节点负载：

$$N_{\text{load}}(t, u) = \frac{CPU(t, u)}{CPU(u)} \quad (1)$$

链路负载：

$$L_{\text{load}}(t, e) = \frac{BW(t, e)}{BW(e)} \quad (2)$$

其中， $CPU(t, u)$ 表示 t 时刻节点 u 的 CPU 资源使用量， $CPU(u)$ 表示节点 u 的 CPU 资源总量； $BW(t, e)$ 表示 t 时刻链路 e 的带宽使用量， $BW(e)$ 表示链路 e 的带宽总量。通过式 (1) 和式 (2) 的定义，可以进而得出节点均衡度和链路均衡度。

节点均衡度：

$$Balance_N(t) = \sqrt{\frac{\sum_{u \in N^S} (N_{\text{load}}(t, u) - \frac{\sum_{v \in N^S} N_{\text{load}}(t, v)}{|N^S|})^2}{|N^S|}} \quad (3)$$

链路均衡度：

$$Balance_L(t) = \sqrt{\frac{\sum_{e \in L^S} (L_{\text{load}}(t, e) - \frac{\sum_{l \in L^S} L_{\text{load}}(t, l)}{|L^S|})^2}{|L^S|}} \quad (4)$$

将节点均衡度和链路均衡度线性组合，就可以得到可重构服务承载网映射的目标函数：

$$Balance(t_k) = \alpha Balance_N(t_k) + \beta Balance_L(t_k) \quad (5)$$

其中， α ， β 分别代表节点均衡度和链路均衡度的权重，多数情况下节点均衡度和链路均衡度是不会同时达到的，所以本文采用对二者加权组合

的形式来评价整个底层物理网络的负载均衡性； t_k 表示第 k 个可重构服务承载网映射请求到达后的时刻。可重构服务承载网映射的目标就是使式 (5) 达到最小，也即让整个物理网络始终处于负载均衡的状态。

可重构服务承载网映射请求是在线的，可以看成是 M/M/1 排队过程，所以无法提前预知。在这种情况下，对节点负载和链路负载进行递推。

当第 k 个请求到达时：

$$N_{\text{load}}(t, u) = N_{\text{load}}(t^-, u) + \frac{\Delta CPU_k(u)}{CPU(u)},$$

$$\forall u \in \{M_k^N(\hat{u}) \mid \hat{u} \in N_k^v\}$$

$$L_{\text{load}}(t, e) = L_{\text{load}}(t^-, e) + \frac{\Delta BW_k(e)}{BW(e)},$$

$$\forall e \in \{M_k^L(\hat{e}) \mid \hat{e} \in L_k^v\}$$

当第 k 个请求注销时：

$$N_{\text{load}}(t, u) = N_{\text{load}}(t^-, u) - \frac{\Delta CPU_k(u)}{CPU(u)},$$

$$\forall u \in \{M_k^N(\hat{u}) \mid \hat{u} \in N_k^v\}$$

$$L_{\text{load}}(t, e) = L_{\text{load}}(t^-, e) - \frac{\Delta BW_k(e)}{BW(e)},$$

$$\forall e \in \{M_k^L(\hat{e}) \mid \hat{e} \in L_k^v\}$$

当第 k 个请求重构时：

$$N_{\text{load}}(t, u) = N_{\text{load}}(t^-, u) - \frac{\Delta CPU_k(u)}{CPU(u)},$$

$$\forall u \in \{M_k^N(\hat{u}) \mid \hat{u} \in N_k^v\}$$

$$N_{\text{load}}(t, v) = N_{\text{load}}(t^-, v) + \frac{\Delta CPU_k(v)}{CPU(v)},$$

$$\forall v \in \{\hat{M}_k^N(\hat{v}) \mid \hat{v} \in N_k^v\}$$

$$L_{\text{load}}(t, e) = L_{\text{load}}(t^-, e) - \frac{\Delta BW_k(e)}{BW(e)},$$

$$\forall e \in \{M_k^L(\hat{e}) \mid \hat{e} \in L_k^v\}$$

$$L_{\text{load}}(t, e') = L_{\text{load}}(t^-, e') - \frac{\Delta BW_k(e')}{BW(e')},$$

$$\forall e' \in \{\hat{M}_k^L(\hat{e}') \mid \hat{e}' \in L_k^v\}$$

其中， t 和 t^- 分别代表第 k 个可重构服务承载网请求到达、注销、重构之后的时刻和之前的时刻； u 和 v 为该承载网请求所对应的任意物理节点， e 和 e' 为该承载网请求所对应的任意物理链路； $\Delta CPU_k(u)$ 表

示物理节点 u 分配给该承载网请求的 CPU 资源, $\Delta BW_k(e)$ 表示物理链路 e 分配给该承载网请求的带宽资源; M_k^N , M_k^L 分别代表该承载网节点映射和链路映射; \hat{M}_k^N , \hat{M}_k^L 分别代表该承载网映射被重构之后新的节点映射和链路映射。此外, 请求重构是指原有节点/链路映射方案发生改变, 其只会发生在请求达到与注销的时间段之内。

4.2 映射算法与机制

基于前述模型, 得到 RFNet 下可重构服务承载网映射算法, 如图 5 所示。

```

Algorithm 1 Intra-domain RSCNM algorithm
Nload(t-, u)
Lload(t-, e)
OUTPUT:
MN, ML

Step 1: Node Mapping
Ns-cond = φ /* 物理节点候选集合 */
Nv-cond = Nv /* 虚拟节点候选集合 */
ns = φ /* 当前物理节点 */
nv = φ /* 当前虚拟节点 */
for i=1, 2, ..., |Ns| do
    nv = arg maxv∈Nv-cond CNv(v)
    S = {v | CNv(nv) ≤ (1 - Nload(t-, v)) · CPU(v), v ∈ Ns - Ns-cond}
    if S is φ then
        VNM is failed, and return.
    end if
    ns = arg minv∈S Nload(t-, v)
    MN(nv) = ns /* 节点映射 */
    Ns-cond = Ns-cond ∪ {ns}
    Nv-cond = Nv-cond - {nv}
end for

Step 2: Link Mapping
for each l=(u,v) ∈ Lv do
    find the shortest paths satisfied the CLv from MN(u) to MN(v): Paths /* 使用 k-shortest paths 算法[21] */
    if Paths is φ then
        VNM is failed, and return.
    end if
    P' = arg minp∈Paths {Balance(t) when ML(l) = p}
    ML(l) = P' /* 链路映射 */
end for
    
```

图 5 RFNet 下可重构服务承载网映射算法

该算法分为节点映射和链路映射 2 部分。其中, 节点映射的基本思想是将节点资源需求大的虚拟节点映射到节点负载小的物理节点, 其前提是该物理节点的剩余资源满足需求; 链路映射的基本思想

是在满足链路约束的所有最短路径中选择可以使整体均衡度指标最小的路径, 并将其作为当前虚拟链路所对应的物理路径。

此外, 如果映射请求对虚拟节点做地理位置的限制, 那么虚拟节点映射将可能会跨越多个管理域 (如图 6 所示), 映射算法也需要进行相应的扩展, 从而产生域间可重构服务承载网映射算法, 如图 7 所示。

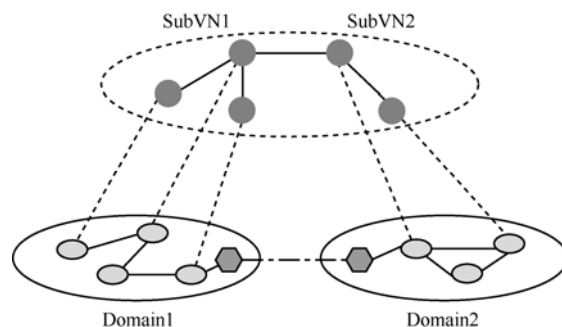


图 6 跨域可重构服务承载网请求的映射

```

Algorithm 2 Inter-domain RSCNM algorithm
INPUT:
Gv = (Nv, Lv, CNv, CLv)
Gs = (Ns, Ls, ANs, ALs)
Nload(t-, u)
Lload(t-, e)
OUTPUT:
MN, ML

Step 1: Partition the VN request
Partition Gv into K subgraphs Gkv = (Nkv, Lkv): /* 根据虚拟节点对所属域的限制 */
Nv = ∪k Nkv
Lv = (∪k Lkv) ∪ LCv
其中 LCv 为跨域的虚拟连接集合
Step 2: Intra-domain VNM
Do Intra-domain VNM Algorithm for Gkv in domain k. /* 域内局部映射 */
Step 3: Establishing inter-connection
for each l=(u,v) ∈ LCv do
    find the shortest path satisfied the CLv from MN(u) to MN(v): Path /* 使用 k-shortest paths 算法[21] */
    Path = (∪i∈D pathi) ∪ Eborder
其中 Eborder 为不同域间的物理连接集合, pathi 为经过域 i 内部的物理路径
    If Path is not found then
        VNM is failed, and return.
    end if
    ML(l) = Path /* 域间虚拟链路映射 */
end for
    
```

图 7 域间可重构服务承载网映射算法

该算法的基本思想是：首先将可重构服务承载网请求根据地理位置约束分割为若干子可重构服务承载网，以及连接不同子可重构服务承载网的虚拟连接；针对每个子可重构服务承载网请求，在相应管理域内施行域内可重构服务承载网映射算法；针对域间虚拟连接，利用最短路径算法得到跨域的物理路径，完成域间虚拟链路的映射。

可重构服务承载网映射是 RFNet 进行网络级重构的关键环节，通过可重构柔性网络管理平台完成，其执行过程如图 8 所示。

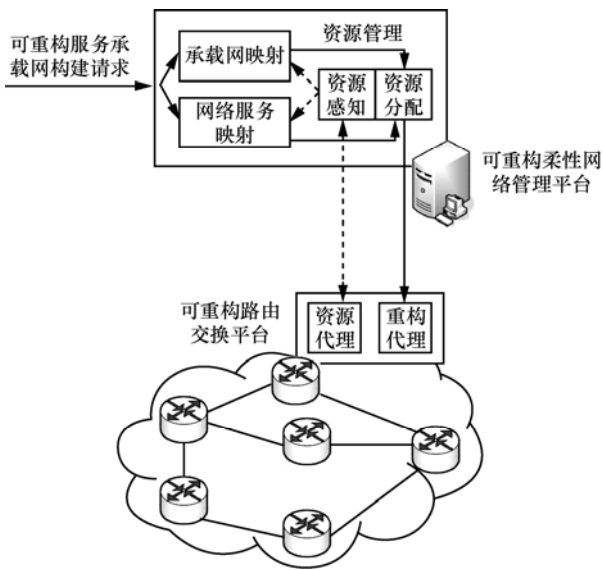


图 8 可重构服务承载网构建过程

从图中可以看出，管理平台通过资源感知模块收集各可重构路由交换平台资源代理模块所反馈的资源使用情况；当有可重构服务承载网构建请求到来时，管理平台结合当前资源情况，分析可重构服务承载网请求，通过可重构服务承载网映射算法和网络服务映射算法得到资源分配策略；之后，通过资源分配模块将资源分配策略转成重构命令，并发送至各可重构路由交换平台的重构代理模块，由其负责对路由器进行实际的重构操作。

5 网络服务映射机制

网络服务映射就是将用户的业务需求映射为构件的过程，即业务—服务—构件的映射过程。网络服务映射过程分为 2 个阶段：拓扑映射和构件组合。拓扑映射阶段主要完成将用户业务类型映射为处理流程图（PFG, process-flow graph）；构件组合阶段主要是根据处理流程图，搜索出最优的构件组

合方案，使其满足总体的业务性能需求。

5.1 拓扑映射

拓扑映射的过程包括业务映射、服务映射和构件映射。

业务映射是对用户提供的业务特性 $T_f = (p_1, p_2, \dots, p_n)$ 进行分类，分类的依据是业务库（TDB, traffic database）中存储的各业务类型的聚类中心 $C_i = (c_1, c_2, \dots, c_n)$ 。聚类中心 c_i 是预先得到的，根据 ATM、DiffServ 等业务聚类标准，对当前网络业务进行聚类，从而得到聚类中心。然后，通过业务分类模块可以得到业务特征 T_f 所属的业务类型索引（TID, traffic type identification）：

$$TID = \arg \min_{1 \leq i \leq n} |T_f - C_i|$$

服务映射是根据业务类型索引查询该业务类型的服务拓扑 G_{serv} 。服务映射库（SMDB, service mapping database）存储了不同业务类型的服务拓扑结构，记为 (TID, G_{serv}) ，其中， $G_{serv} = (Sr, E_{serv})$ ， $Sr = \{S_i | S_i = (SID, Name, Description), i = 1, 2, \dots, k\}$ 为服务单元的集合， E_{serv} 为服务单元间的连接关系。整个 SMDB 相当于专家知识库，每种业务类型在功能上可以分解为由若干服务单元组成的拓扑结构，这些结构作为专家知识被存储和管理。服务映射代理的作用是根据业务类型索引对 SMDB 进行查询，返回相应的服务拓扑结构。

构件映射是根据服务单元索引（SID, service unit identification）查询该服务单元所对应的处理流程（process-flow），从而将服务拓扑 G_{serv} 映射为处理流程图 G_{proc} 。构件映射库（CMDB, component mapping database）存储了不同服务单元所对应的处理流程，记为 (SID, G'_{proc}) ，其中：

$$G'_{proc} = (Pr, PID_{source}, PID_{sink}, E_{proc})$$

$Pr = \{P_i | P_i = (PID, Name, Description), i = 1, 2, \dots, k\}$ 为构件处理集合， E_{proc} 为构件处理间的连接关系， PID_{source} 为该处理流程图的源节点索引， PID_{sink} 为该处理流程图的末节点索引。服务单元分解成处理流程的方案作为一种专家知识被 CMDB 存储和管理，从而使得 CMDB 成为一个专家知识库。构件映射代理则根据服务单元索引 SID 对 CMDB 进行查询，得到各自对应的处理流程图 G'_{proc} ，最后将之组合成整个业务类型的处理流程图 G_{proc} 。

5.2 构件组合

拓扑映射阶段生成了处理流程图 G_{proc} ，其每个节点均为一个处理单元 P_i ，根据 P_i 的描述信息 **Description** 对构件库进行基于关键字的检索操作，可以得到满足 P_i 处理能力的候选构件集合。构件组合阶段就是从每个候选构件集合 i 中挑选出第 j 个构件 C_{ij} ，从而形成构件组装图 G_{comp} 。

构件组合的依据是性能。由于 RFNet 下的网络服务在运行时需要满足一定的性能要求 $Cap_{req}=[p_1, p_2, \dots, p_n]$ ，所以候选构件都具备性能模型。候选构件 C_{ij} 的性能模型可以描述为

$$Cap_{ij} = (Cap_{in}, Res, Cap_{out})$$

其中， $Cap_{ij} = [p_1^{in}, p_2^{in}, \dots, p_n^{in}]$ 表示构件 C_{ij} 所需要的输入性能属性矢量； $Cap_{out} = (p_1^{out}, p_2^{out}, \dots, p_m^{out})$ 表示构件 C_{ij} 所提供的输出性能属性矢量； $Res = [r_1, r_2, \dots, r_k]$ 则为构件 C_{ij} 运行时所占用的系统资源矢量。此外，性能属性包含名称和取值约束 2 部分，即 $p_i = (Name_i, Value_i)$ ，其中， $Value_i$ 表示取值集合（包含单值）。

构件组合问题建立在整数线性规划模型基础上，其复杂性为 NP-hard 问题。如果采用枚举算法，则计算时间复杂度为 $O(M^n)$ ，所以仅适用于小规模构件组合。对于大规模构件组合，本文采用模拟退火算法进行近似求解。

5.3 节点重构机制

可重构路由交换平台的体系结构基于开放式、转发与控制相分离的思想实现，支持路由交换节点级的重构，并定义了软硬件方面的完整重构模型。可重构路由交换平台的重构包括硬件重构和软件重构，其中软件重构过程如图 9 中 A 箭头所示，硬件重构过程如 B 箭头所示。

当主控部件接收到来自可重构柔性网络管理平台的重构命令时，重构代理模块负责检测并解析命令，如果涉及到软件重构，则交由软件构件代理来处理，反之则交由硬件构件代理。对于前者，软件构件代理进行如下操作：1) 接收软件重构描述文件，并提取相关信息；2) 判断所需的构件在构件库系统中是否齐全，并提取其描述信息；3) 判断可重构路由交换平台的资源是否足够此次重构；4) 从构件库系统中下载所需的软件构件；5) 将软件构件部署到软件构件运行环境中，并启动构件；

6) 返回重构结果。对于后者，硬件构件代理进行如下操作：1) 接收硬件重构描述文件，并提取相关信息；2) 判断所需的构件在构件库系统中是否齐全，并提取其描述信息；3) 根据单板资源管控模块提供的信息，判断单板资源是否足够此次硬件重构；4) 从构件库系统中下载所需的硬件构件；5) 将硬件构件发送至相应的单板部件中，再通过单板资源管控模块将其部署到硬件构件运行环境中；6) 返回重构结果。

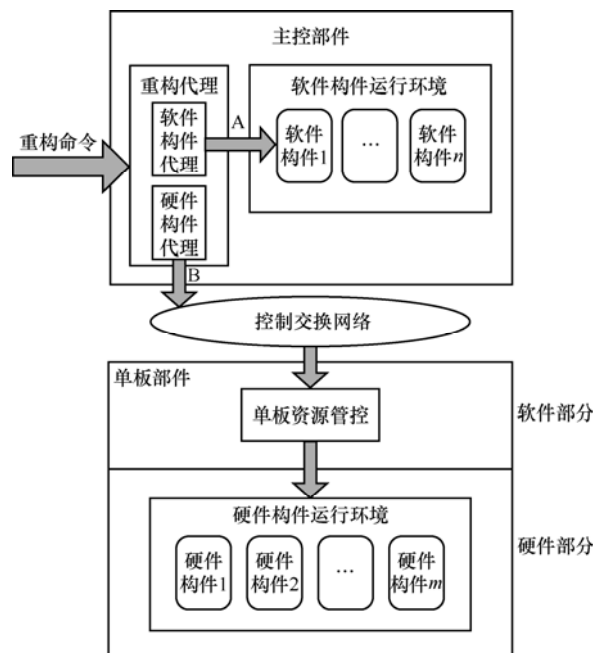


图 9 可重构路由交换平台的重构过程

主控部件 (CE) 是可重构路由交换平台的控制核心，主要负责路由平台的业务配置、运行管理、资源监控、路由计算、路由下发、软硬件构件管理等工作，其软件架构如图 10 所示，包括：管理接口、构件运行环境与构件、构件管理与监控、服务映射层、设备抽象层、ForCES 协议中间件、Linux 操作系统。

其中，管理接口为外界提供资源代理和重构代理等模块，使可重构柔性网络管理平台能对可重构路由交换平台进行资源管理和重构；构件运行环境为软件构件提供生命周期内所需的各类资源，如 CPU、内存等资源；主控部件中的构件包括路由协议构件、基本协议栈构件、管理构件、网络安全构件、QoS 构件等；构件管理与监控模块主要完成构件的动态加载/卸载、实例化、析构、绑定和解除绑定接口，以及在此基础上建立和约束动态重构的环

3) 平均链路利用率

平均链路利用率是衡量利用算法所构建的服务承载网络所占链路带宽之和与物理网络分配的所有链路资源带宽之和的比值。

$$\bar{U} = \frac{\sum_{\forall e^{LCN} \in e^s} B_{e^{LCN}}}{\sum_{e \in e^s} B_e}$$

该式所定义的平均链路利用率可以衡量算法对整网链路资源的均衡利用情况。

4) 构建平均收益

构建 RSCN 是服务提供商通过“批发”物理网络资源，“零售”给各业务提供商。因此，应该定义指标来衡量构建算法带来的收益。本文所定义的构建收益是服务提供商构建 RSCN 后，形成服务能力卖给业务提供商所获得的收益，与业务提供商所需的 RSCN 带宽 $bw_i(l^v)$ 成正比。

$$R(G^v(t)) = \sum_{i \leq I} \sum_{l^v \in L^v} bw_i(l^v)$$

而构建平均收益定义为

$$\bar{R} = \frac{\sum_{t=0}^T R(G^v(t))}{T}$$

表示一段时间内网络构建收益的平均值。

6.3 结果讨论

1) 构建成功率

图 11 横轴表示的请求数为从仿真起始时刻开始到本时刻，累计到达的所有请求的个数。由仿真结果可以看出，当建网请求到达数不多时，几种算法的构建成功率都可以接近或等于 100%，但随着请求数逐渐增加，由于各种算法资源分配方法的不同，使构建成功率出现了不同的变化，接近线性的下降过程。而随着请求数的进一步增加，原来存在的 RSCN 达到生存时间，不断有网络资源释放，因此，构建成功率会逐渐达到一种统计上的稳态。在比较的 4 种算法中，文献[14]中的 basic-VN 和 sub-VN 算法采用启发式方法进行节点映射，且假设资源是无限的，为性能比较，改动为适合本文仿真环境。由仿真结果发现，构建成功率基本为 50% 左右，并没有 RSCNM 算法的构建成功率高。这是因为 basic-VN，sub-VN 虽然考虑了节点可用资源及最大负载强度，但链路映射仍然采用与文献[13]相似的最短路径选路算法 SPF，容易产生瓶颈链路；

RSCNM 构建成功率高于 60%，由于 RSCNM 算法不使用最短路径进行选路，并且在资源分配过程中均衡了负载强度，使每一次的分配给后续的构建造成的影响相对都是均衡的，使构建成功率明显要高；可以看出 RSCNM 构建成功率比其他算法有 10% 以上的提高。

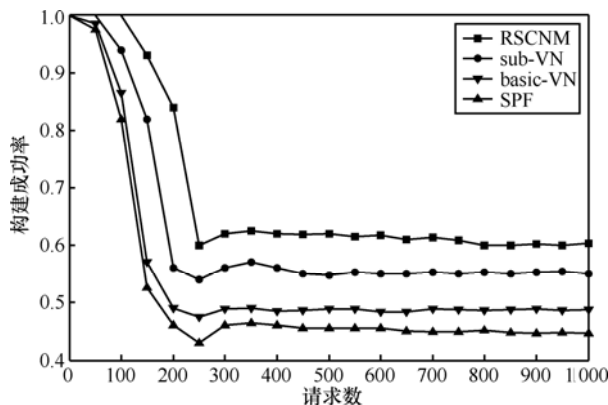


图 11 构建成功率

2) 平均链路利用率

由图 12 可以看出，请求数较少时，几种算法所带来的平均链路利用率都是快速上升的，随着请求次数增加到 170 次左右，在每个时间窗内不断有 RSCN 到达生存期，释放资源，使平均链路利用率达到一种统计上的稳态。由于 RSCNM 算法引入了负载强度控制，刚开始链路负载分布较均衡，平均链路利用率并不高，当其他算法已经出现较高平均链路利用率后，RSCNM 算法在请求次数达到 200 次才开始升高，并且均衡分布负载，构建成功率要高于其他算法，当达到统计稳态后，平均链路利用率比其他算法都要高。从图 12 还可以看出，由于 RSCNM 算法根据物理网络状态自适应选择偏重链路强度均衡的路径或者节点强度均衡的路径，可以看出不管是偏重于哪个指标，对链路利用率的影响都是积极的，使得链路资源使用均衡，因此最终得到的平均链路利用率是相对较高的。

3) 最大节点强度

由图 13 可以看出，对于到达的请求，RSCNM、basic-VN、sub-VN 算法稳态的最大节点强度接近 45%，靠近平均节点强度 40%，比起 SPF (55%) 来说，有了较大的改善。这是因为 RSCNM 算法包含了根据路径平均节点强度进行计算的步骤，而 basic-VN、sub-VN 在节点映射时采用了最小化最大

节点强度的算法。同时，从图 13 还可以看出，一定的到达请求次数内，RSCNM 的最大节点强度略比 basic-VN，sub-VN 要好，但有时受到算法调整链路强度影响而发生变化。这是因为 RSCNM 算法强调网络资源分配的均衡性，使每个节点上分布的 RSCN 个数相对均衡，并不会为某次资源分配而造成网络负载失衡；并且采取自适应调整，有时会根据链路强度进行调整，因此会对它的最大节点强度性能有所影响。

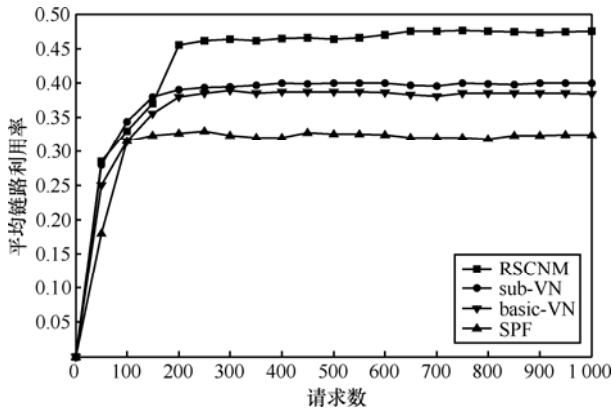


图 12 平均链路利用率

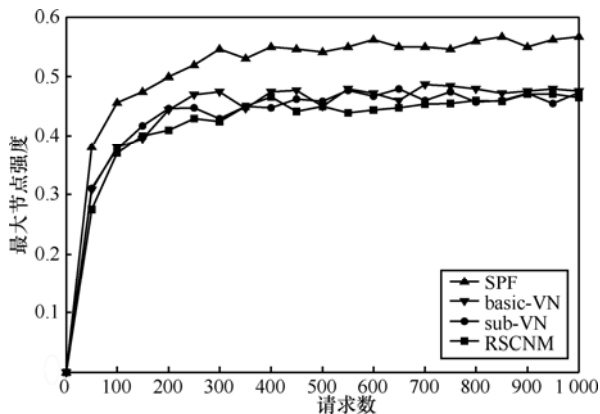


图 13 最大节点强度

4) 构建平均收益

由图 14 可以看出，根据 RSCN 构建平均收益的定义，随着请求数增加，由于能够构建的 RSCN 数目增加，收益呈递增的趋势。当请求数到达一定数量后，由于网络可用资源逐渐减少(释放资源相对构建请求所需资源，还远远不够)，构建成功率的降低并呈稳态，能够构建的 RSCN 数目也呈稳态趋势，构建平均收益也呈稳态分布(由于每隔一段时间释放一些资源，因此，尽管构建成功率降低，但存在的 RSCN 数目及所用资源基本成稳态趋势)。

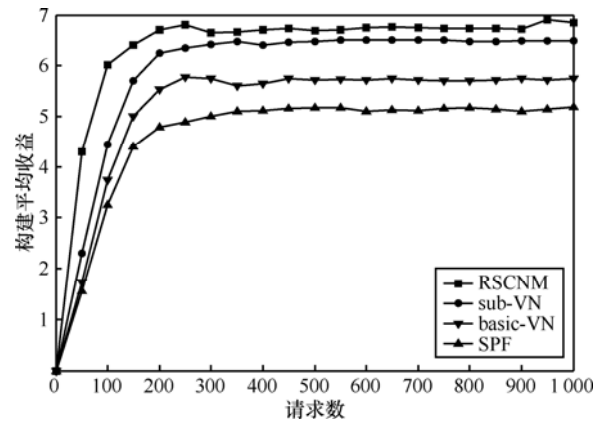


图 14 构建平均收益

7 结束语

本文在可重构柔性网络体系结构背景下，对“资源管理—承载网映射—网络服务映射”机制进行了详细论述。首先，介绍了可重构柔性网络的体系结构；其次，阐述了可重构柔性网络的资源管理机制；然后，阐述了可重构服务承载网映射机制，并着重提出一种基于负载均衡的可重构服务承载网域内、域间构建算法；最后，根据可重构服务承载网的资源映射方案，提出一种基于构件化可重构路由交换平台的网络服务部署机制。通过仿真实验验证了本文提出的可重构服务承载网映射算法在网络构建成功率、最大节点强度、平均链路利用率和构建平均收益上具有明显的优势。

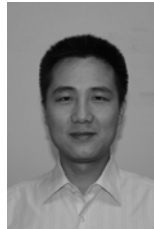
本文提出的可重构柔性网络“资源管理—承载网映射—网络服务映射”流程机制已成功应用在建成的“可重构柔性实验网”之上。未来将通过部署更多的业务应用来测试、验证和完善所提出的关键机制。

参考文献：

- [1] TURNER J, TAYLOR D. Diversifying the internet[A]. Proceedings of the IEEE Conference on Global Telecommunications[C]. St Louis, USA, 2005. 755-760.
- [2] GENI. Global environment for network innovations[EB/OL]. <http://www.geni.net>.
- [3] FIND. Future internet network design[EB/OL]. <http://find.isi.edu>. San Jose, USA, 2004.
- [4] FIRE: future internet research and experimentation[EB/OL]. <http://cordis.europa.eu/fp7/ict/fire>, 2010-12-29.
- [5] 4WARD[EB/OL]. <http://www.4ward-project.eu/>.
- [6] NICT. New Generation Network Architecture — AKARI Conceptual Design (ver1.1)[S]. October 2008.

- [7] FP7[EB/OL]. <http://cordis.europa.eu/fp7>.
- [8] HU Y X, LAN J L, WU J X. Providing personalized converged services based on flexible network reconfiguration[J]. *Science China Information Sciences*, 2011, 54(2): 334-347.
- [9] 李玉峰, 邱菀, 兰巨龙. 可重构路由器研究的现状与展望[J]. *中国工程科学*, 2008,10(7):82-95.
LI Y F, QIU H, LAN J L. Status quo and outlook of reconfigurable research[J]. *China Engineering Sciences*, 2008, 10(7): 82-95.
- [10] ANDERSON T, PETERSON L, SHENKER S, *et al.* Overcoming the Internet impasse through virtualization[J]. *Computer*, 2005, 38(4): 34-41.
- [11] HAIDER A, POTTER R, NAKAO A. Challenges in resource allocation in network virtualization[A]. *Proceedings of the ITC Specialist Seminar on Network Virtualization*[C]. 2009:131-140.
- [12] Ricci R, ALFELD C, LEPREAU J. A solver for the network testbed mapping problem[J]. *ACM Computer Communication Review*, 2003, 33(2):65-81.
- [13] YU M, YI Y, REXFORD J, *et al.* rethinking virtual network embedding: substrate support for path splitting and migration[J]. *ACM SIGCOMM Computer Communications Review*, 2008,38(2):17-29.
- [14] ZHU Y, AMMAR M. Algorithms for assigning substrate network resources to virtual network components[A]. *Proceedings of the IEEE INFOCOM*[C]. Barcelona, Catalunya, Spain, 2006. 1-12.
- [15] LU J, TURNER J. Efficient Mapping of Virtual Networks onto a Shared Substrate[R]. WUCSE-2006-35, Washington University, 2006.
- [16] CHOWDHURY N, RAHMAN M, BOUTABA R. Virtual network embedding with coordinated node and link mapping[A]. *Proceedings of the IEEE INFOCOM*[C]. 2009. 783-791.
- [17] LISCHKA J, KARL H. A virtual network mapping algorithm based on subgraph isomorphism detection[A]. *Proceedings of the ACM SIGCOMM VISA*[C]. 2009. 81-88.
- [18] HE J, ZHANG S R, LI Y, *et al.* Davinci: dynamically adaptive virtual networks for a customized internet[A]. *Proceedings of the ACM CoNEXT Conference*[C]. Madrid, SPAIN, 2008. 1-12.
- [19] RAHMAN M R, AIB I, BOUTABA R. Survivable virtual network embedding[A]. *Proceedings of the 29th Conference on Computer Communications (IEEE INFOCOM)*[C]. 2010. 40-52.
- [20] CAI Z P, LIU F, XIAO N, *et al.* Virtual network embedding for evolving networks[A]. *Proceedings of the IEEE Global Telecommunications Conference*[C]. 2010. 1-5.
- [21] EPPSTEIN D. Finding the k shortest paths[A]. *Proceedings of the IEEE Symposium on Foundations of Computer Science*[C]. 1994. 154-165.
- [22] ZEGURA E W, CALVERT K L, BHATTACHARJEE S. How to model an internetwork[A]. *Proceedings of the IEEE INFOCOM*[C]. 1996. 594-602.

作者简介:



王保进 (1974-), 男, 河北保定人, 国家数字交换系统工程技术研究中心讲师, 主要研究方向为可重构柔性网络、宽带信息网络、嵌入式系统等。



汪斌强 (1963-), 男, 安徽安庆人, 国家数字交换系统工程技术研究中心教授、博士生导师, 主要研究方向为宽带信息网络、可重构柔性网络、高速路由器核心技术等。



王志明 (1986-), 男, 辽宁大连人, 国家数字交换系统工程技术研究中心硕士生, 主要研究方向为可重构路由交换技术、网络虚拟化技术等。