

面向内容中心网络基于分布均匀度的蚁群路由策略

张国印, 唐滨, 孙建国, 李佳楠

(哈尔滨工程大学 计算机科学与技术学院, 黑龙江 哈尔滨 150001)

摘要: 面向内容中心网络, 采用转发服务评价函数得到转发端口集合, 结合端口集合和自适应转发策略提出一种基于分布均匀度的层次化蚁群算法, 通过感知路由信息素来选择最佳路径, 综合考虑收敛时间和精确性对算法进行优化。仿真实验表明, 所提出算法在降低内容请求时延和均衡节点负载方面有较好的表现。

关键词: 内容中心网络; 转发策略; 蚁群算法; 分布均匀度

中图分类号: TP302.1

文献标识码: A

Ant colony routing strategy based on distribution uniformity degree for content centric network

ZHANG Guo-yin, TANG Bin, SUN Jian-guo, LI Jia-nan

(College Computer Science & Technology, Harbin Engineering University, Harbin 150001, China)

Abstract: By combining port set with adaptive forwarding strategy, an ant colony algorithm based on hierarchical distribution uniformity was proposed. Through perceiving routing pheromone, the optimal path was selected. At the same time, the algorithm was optimized by considering both convergence time and precision. Simulation results show that the algorithm has better performance in terms of the delay of content request and node load balance.

Key words: content centric network; forwarding strategy; ant colony algorithm; distribution uniformity degree

1 引言

随着互联网的快速发展, 网络模型和用户需求间不匹配的矛盾日益突出。另外, TCP/IP 协议在移动性、安全性、可靠性和灵活性上也暴露了诸多问题。学术界提出了众多的革命性和创新性的解决方案, DONA (data-oriented network architecture) 是由美国伯克利大学 RAD 实验室提出的以信息为中心的网络体系架构。DONA 对网络命名系统和名字解析机制做了重新设计, 替代现有的 DNS, 使用扁平结构, 然后使用彼岸准的 IP 路由和转发进行数据分组的传递。PSIRP (publish-subscribe Internet routing paradigm)^[1,2] 旨在建立一个以信息为中心的发布-订阅通信范例, 取代以主机为中心的发送-接收通

信模式。PSIRP 改变路由和转发机制, 完全基于信息的概念进行网络运作。作为 CCN 的一个典型范例, NDN (named data networking)^[3,4] 的提出改变了当前互联网主机-主机通信范例, 使用数据名字而不是 IP 地址进行数据传递, 让数据本身成为因特网架构中的核心要素。NDN 基于名字的路由和转发解决了 IP 网络中地址空间耗尽、NAT 穿越、移动性和可扩展的地址管理 4 个问题。

内容中心网络 (CCN, content-centric-network) 有 2 种类型的数据分组: Interest 和 Data, 解耦数据与位置的关系, 当一个数据分组到达 Face 的时候, 利用名字最长前缀匹配原则。传统的路由协议 (如 OSPF、IS-IS、BGP) 适用于基于名字前缀的路由。当有多个兴趣分组同时请求相同数据时, 路由器只

收稿日期: 2014-04-26; 修回日期: 2014-08-20

基金项目: 国家自然科学基金资助项目 (61073042, 61003036); 黑龙江省自然科学基金资助项目 (F201121); 中央高校基本科研业务费专项基金资助项目 (HEUCF100606)

Foundation Items: The National Natural Science Foundation of China (61073042, 61003036); The Natural Science Foundation of Heilongjiang Province (F201121); Fundamental Research Funds for the Central Universities (HEUCF100606)

会转发收到的第一个兴趣分组,并将这些请求存储在PIT中。当数据分组传回时,路由器会在PIT中找到与之匹配的条目,并根据条目中显示的接口列表,分别向这些接口转发数据分组。转发完成后,路由器会删除相应的PIT条目,并将数据分组存储在Content Store中。

在内容中心网络中,一个值得研究的问题是如何对网络路由算法进行优化。内容中心网络根据请求兴趣分组内容名称前缀进行最长匹配的路由机制提高了数据传输的效率,但多路转发特性也造成了网络内容请求冗余。在CCN中,原有的路由机制在基于内容名称的路由和网络的内容分布高速动态环境中无法发挥CCN内在的性能优势。

本文提出一种基于服务的CCN蚁群路由选择策略。首先,对于CCN的转发策略进行建模,提出服务质量函数来表示服务质量;其次在服务质量满足的转发接口集合上使用分布均匀度的蚁群算法,根据转发接口列表信息素分布的均匀度动态调整蚁群算法的收敛速度,当转发层面探测出链路失效或网络拥塞等问题时,转发层面会向下游请求节点发送NACK(negative acknowledgement)兴趣分组进行反馈,使路由算法能够快速调整最佳转发接口。

2 国内外研究现状

“以信息为中心”的概念是未来互联网研究普遍采用的思想^[3]。TRIAD^[5]、DONA^[6]、DOT、PSIRP^[1]、CCN^[7-9]等对此进行了有益的探索和研究。2010年,美国国家科学基金会NSF(national science foundation)资助了4个未来互联网体系架构FIA(future internet architecture)项目,分别是:命名数据网络(NDN, named data networking)、MobilityFirst、NEBULA和XIA(expressive Internet architecture)^[10]。

在国内,未来互联网体系架构的研究也受到各大高校和研究机构的广泛关注。国家自然科学基金委员会先后启动了“未来互联网体系架构理论研究”、“新一代网络应用平台及其管理的基础理论与关键研究技术”和“网络综合实验平台”等研究课题。清华大学等单位承担了“未来互联网体系架构的理论研究”项目,北京大学建立了除美国本土外全球唯一一个内容中心网络实验节点,实现了与国际新一代网络的互联。

路由转发机制作为CCN核心部分之一,是整

个CCN体系架构重点研究方向。Lixia Zhang团队提出CCN路由的设计目标:可扩展性、安全性、健壮性和高效性。CCN的转发过程可以适应基于路由表和数据层面两者输入产生的转发决策。转发策略的设计目标为:高效地使用网络资源、稳健的分组传输和减少网络开销。CCN的转发模式主要包括3类数据结构,分别是转发信息(FIB, forwarding information base)表,内容存储(CS, content store)表以及未决请求(PIT, pending interest table)表。其中FIB保存了路由到达内容服务器下一跳的接口,CS保存缓存内容,PIT中记录未得到相应报文的名字信息及其到达Face,便于Data报文沿途返回。

CCN的路由技术和IP网络的路由技术类似,都是通过前缀匹配来选择下一跳的转发接口。但CCN的前缀是变长的,而IP网络的前缀是定长的。所以,CCN是IP网络的一个超集,现有IP网络的路由解决方案均可以运用到CCN路由中。针对兴趣分组没有包含源地址和目的地址的情况,文献[10]提出了一种基于原有OSPF协议改造的基于名字的路由信息协议,OSPFN扩展OSPF进行内容前缀的分发。针对CCN的特点,文献[11, 12]提出基于等级的路由机制,文献[13]提出了一种基于GPU加速的方式来实现内容名称的快速查找,文献[14]提出了命名数据网络的邻居缓存路由策略,文献[15]提出了一种基于ISP聚合的路由优化算法。

FIB表中每个名称前缀关联一个转发接口列表,可能具有多个可用转发接口。转发策略是根据接口列表选择转发接口的方法。CCN现有的转发策略主要有:全转发策略、随机转发策略和蚁群转发策略。CCN默认采用全转发策略,此策略获取数据分组时延小,但容易导致请求冗余。随机转发策略不会产生冗余的网络流量,但是无法保证获得稳定和高效的网络性能。

文献[16]针对路由协议的content store设计,使用推拉协议进行内容的检索,但在转发层面缺少必要的考虑。Shashank S^[17]等根据CCN和SCN(service-centric networking)的特征提出了基于蚁群优化算法的路由策略SoCCeR,在CCN的上层增加了路由决策层来对FIB进行操作,模拟结果显示在引入较轻负载的情况下可得到较高的性能提升。使用蚁群算法来辅助决策路由,针对CCN内在不支持服务的特点,提出一种分布式的SCN的架构、分布式的服务选择机制、考查服务负载和路径吞吐

量的网络要素。但对于蚁群算法获得的信息素的浓度没有考虑其层次关系问题, 对如何界定信息素的浓度缺少统一考虑。文献[18]在 CCN 中引入了层次化的设想, 对于 CCN 路由系统的不确定性引入了内容熵的概念。内容熵随着 CCN 路由数量、节点数量的增加而增加。DHR-CCN 将 CCN 划分为 2 层, 核心名字交换网络 (CNSN) 和边名字交换网络 (ENSN), CNSN 由超级内容路由器组成, ENSN 由边内容路由器组成。提出了构成方法和路由算法, 借鉴了 P2P 中超级节点的思想划分 CCN 的节点。但构造节点划分带来的代价是不可忽视的, 单纯从节点类型的考虑将极大影响 CCN 本身的内容检索设计, 在内容的流行度和节点的构造之间也缺少必然的联系。

蚁群算法是一种有效的搜索算法, 具有正反馈特性、分布式特性、自适应性、交互性和顽健性等特点。蚁群算法用在网络路由的基本思想为: 首先创建一个信息素表, 记录每个转发接口信息素的浓度, 通过信息素的浓度得出接口的转发概率; 然后每个网络节点以固定时间间隔随机选择一个网络节点发送探测蚂蚁, 该蚂蚁用于收集网络节点和网络链路的状态信息。当探测蚂蚁到达目的网络节点时, 更新该节点信息素表中目的网络地址为该蚂蚁源地址记录的信息素浓度值, 再通过信息素浓度值计算出其对应的概率值。结合蚁群算法对 CCN 的路由机制的研究, 近年内获得了较多关注^[14, 19-21]。其基本的思路是将网络划分为多个域, 采纳基于域的贪心蚁群路由算法来支持智能的分组传递, 研究中使用 2 类蚂蚁: 一类是域内; 另一类是域间。在动态网络中, 提高了服务质量, 评估实验表明成本效率和伸缩性均有提高。文献[22]根据优化过程中解的分布均匀度, 采用蚁群算法自适应地调整路径选择概率的确定策略和信息量更新策略。然而, 就算法本身而言, 虽然蚁群路由算法具有较好的优化特性, 但其时间复杂度较高, 容易造成蚁群在最初阶段求解收敛速度慢, 产生早熟停滞现象。即搜索进行到一定程度后, 容易因为次优解的信息素被不断加强而使蚂蚁被吸引到一个次优路径, 从而造成算法过早收敛而得不到真正的最优解。从 CCN 的角度出发, 一个内容匹配的转发过程不仅要考虑节点接口的信息素浓度以及相应的转发概率, 更应该从服务本身出发, 传输时间和网络负载作为衡量网络的重要参数, 预留必要的网络资源来保证关键服

务得到优先满足。国内外对于网络服务质量的研究已经取得了一定成果^[23-26], 但是对于 CCN 的服务质量问题研究国内外还少有提及。

3 CCN 的自适应转发策略

自适应转发策略的主要目标是为了实现通过最佳路径检索数据和快速探测出数据分组传输的问题并恢复。路由器会在其 PIT 数据结构中记录每一个已经转发出去的兴趣分组及其转发接口, 称其为“数据报状态”。

当一个兴趣分组转发出去时, 路由器会启动一个定时器。如果在定时器到期时仍然没有收到数据分组, 则说明通过该接口的网络链路上可能存在问题。但是, 基于定时器这种机制来探测问题是十分耗时的, 因为它必须等到定时器结束才能断定最后结果。更糟糕的是, 当检索的路径上没有发现数据时, 待定兴趣分组仍然保留在网络中直到自身的生存时间到期。这潜在地阻塞了其他请求同一数据的兴趣分组, 因为 PIT 条目中已存在该兴趣分组的转发记录。为了解决这些问题, 自适应转发策略引入一种否定应答 NACK 兴趣分组。当节点不能提供数据分组或转发兴趣分组时, 该节点则向下游节点发送一个 NACK 兴趣分组作为反馈。NACK 兴趣分组除了包含相同请求内容的名称, 并添加一个错误代码来解释 NACK 产生的原因。NACK 兴趣分组可以快速地将网络出现的问题反馈给下游节点, 减少了其等待时间。

针对每一个内容名称, PIT 表中都存在一个记录条目。该条目包含请求接口列表和兴趣分组转发接口列表。通过 PIT 表来维持数据报的转发状态。同样, FIB 表也针对每个内容名称创建一个条目, 每个条目都具有一个按照网络性能排序的转发接口列表。每一个转发接口都具有自身的接口标识、路由性能、往返时延和速率限制。接口等级可以通过多种算法进行确定。另外, 需要确定每个接口的速率限制, 当超过最大数据速率应该避免从该接口进行转发, 防止网络拥塞的发生。

设 L_i 为接口 i 的最大数据速率, C_i 为接口 i 网络上链路容量, \bar{S}_i 为通过接口 i 接收到的数据分组的估计值, 则 $L_i = \alpha C_i / \bar{S}_i$, α 为配置参数, 用于在计算中对网络问题的补偿, 如: 不精确的数据大小、链路或网络层面的开销。当接口 i 传输速率达到 L_i 时, 节点将不能通过该接口进行转发。

根据 PIT 表和 FIB 表存储的数据报状态, 策略模块决定通过哪个接口转发兴趣分组, 使转发决策动态地适应于网络条件。当路由器接收到一个兴趣分组并且 PIT 表中不存在时, 路由器通过 FIB 表中与之匹配的接口列表的第一个接口进行转发, 并创建一个 PIT 条目用于记录这一兴趣分组及其转发接口。当路由器接收到一个 NACK 兴趣分组时, 说明通过上一个接口转发失败, 所以选择下一个等级接口转发兴趣分组。理想情况下, 路由器应该在不需花费太长的时间内尝试多个转发接口, 直到成功接收到数据分组或超时。当超时后, 路由器将会停止尝试其他接口并向下游节点发送 NACK 兴趣分组, CCN 转发模型如图 1 所示。

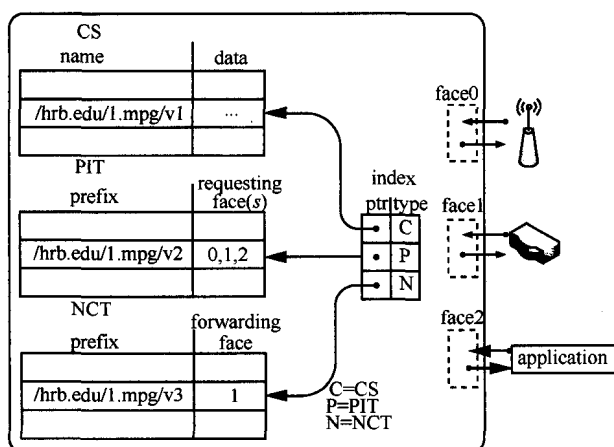


图1 CCN 转发模型

4 基于服务保证的转发优化模型

在一次转发过程中必然是有些端口参与转发而有些不参与。定义决策变量 x_i 表示节点 i 是否参与数据转发, 取 1 表示参与转发, 0 表示不参与转发。表述如下

$$x_i = 1, 0 \quad (1)$$

令一次转发任务选定的端口节点个数为 k , 节点为 $N_i (0 \leq i \leq k-1)$, 各端口对应一次数据转发请求的平均转发速度为 $a_i V_i (0 \leq i \leq k-1)$ 。理论上从各节点转发的数据量 m_i 占总数据量的比例为

$$m_i = a_i V_i / V_a \quad (2)$$

数据传输越长, 则对网络负载的影响越大, 所以要度量不同路径上转发的数据量与该路径的长度的关系。在数据传输过程中, 从节点 i 所能转发的平均数据量可表示成 $a_i V_i / V_a$ 。其中, V_a 表示一次

数据转发中参与服务的端口节点所能提供的平均速度和。

$$V_a = \sum_{i=1}^k x_i a_i V_i \quad (3)$$

网络带宽有一定的阈值限制(q), 而且为了保障其他数据服务得到满足(创建连接), 每个节点可以连接的端口节点数必须有限制。阈值可根据正常情况下的网络接入带宽或可以达到的最大速度进行设定。则约束方程如下

$$\sum_{i=0}^k x_i V_i \leq q \quad (4)$$

在满足转发时间约束的前提下, 尽可能地降低转发时间, 并尽量少地建立与各端口节点的连接, 降低整体网络的负载。令一次数据服务的转发时间 t 满足如下范围约束

$$t_l < t < t_r \quad (5)$$

定义 1 (服务质量函数) 一次数据转发的期望传输完成时间为 t_l , 最长时间为 t_r , 数据服务的重要等级为 z , 服务请求者对转发时间的期望度为 r , 那么服务质量函数 f 可以表示成函数 $F(r, t, z)$, F 为 3 个变量的增函数。

F 的一种较为简单形式可以表示为

$$f = r(t - t_e)^z \quad (6)$$

很显然, f 是 t 的增函数, t 值越大相应的 f 值越高, 说明 QoS 越差, 因此应在允许的情况下确保 f 的值偏小。同样, 网络负载 λ 越大说明网络代价越高。那么, 可以对 f 和 λ 同时进行优化, 并根据网络整体状况设定权重因子 π , 均衡两者的取值。

转发优化模型为

$$\min_x Z = \pi \mu f + \lambda; \quad \text{s.t.: } \{(1), (2), (3), (4), (5)\} \quad (7)$$

其中, f 与 λ 均为决策变量 x 的函数, μ 为权重系数, $\min_x Z$ 表示 Z 为 x 的函数。约束条件 s.t. 为大括号内的几个方程。该 0-1 整数规划模型的意义为: 在满足 s.t. 约束时, 取 Z 的最小值, 由此求解出决策变量 x_i 的取值, 即参与转发的全部节点集合。

蚁群具有初始化时期收敛速度慢和搜索到一定程度后容易产生停滞现象的缺点。而加速收敛速度会造成容易产生次优路径就停滞的问题, 避免停滞现象又会出现收敛速度慢的缺陷。因此, 通过本节的节点集合构造, 将构造一个综合优化的节点集合, 通过对该集合进行分布均匀度的蚁群路由, 得到进一步优化的转发机制。

5 基于分布均匀度的CCN蚁群路由优化算法

根据蚁群算法本身存在的收敛速度慢和容易产生停滞现象的问题,结合内容中心网络本身体系架构具有的优势,基于上节中的节点输出集合,设计了一种基于分布均匀度和转发策略相结合的CCN蚁群路由优化算法,即RED-ACO (content centric networking routing algorithm based on ant colony optimal with equilibrium of distribution)。

5.1 相关概念

基于分布均匀度的蚁群算法相比于其他改进的蚁群算法,该算法根据蚁群在路径上的分布均匀情况动态地调整信息素的更新策略和可选路径的数目及其概率。通过这种动态调整来达到加快收敛速度和防止出现停滞现象这对矛盾的平衡。

定义2 (聚度) 令某网络中存在网络节点 i ,节点 i 的FIB表中存在着许多内容名称对应的转发接口列表。设某具体内容名称 c 对应的转发接口个数为 N_i^c ,每个接口 j 对应信息素的值为 $\tau_{i,j}^c(x)$,通过归一化使每个接口信息素的总和为1,即 $\sum_{j=1}^{N_i^c} \tau_{i,j}^c(x) = 1$ 。 $E_i^c(x)$ 表示所有匹配内容名称 c 接口信息素值的期望值, $S_i^c(x)$ 表示节点 i 中内容名称为 c 聚度的值,则

$$E_i^c(x) = \frac{\sum_{j=1}^{N_i^c} \tau_{i,j}^c(x)}{N_i^c} \quad (8)$$

$$S_i^c(x) = \sqrt{\sum_{j=1}^{N_i^c} (E_i^c(x) - \tau_{i,j}^c(x))^2} \quad (9)$$

分析 为了衡量蚁群在路径上的分布均匀度,引入了“聚度”的概念。当路径上的蚂蚁分布比较分散时,聚度则比较小;当蚂蚁集中在若干条路径上时,聚度则比较大。聚度较小时,说明蚁群之间还没有形成最优路径,需要加快收敛速度,使较优的路径以较大的概率被选择,并较大程度地增加相应路径信息素的值;聚度较大时,易出现早熟和停滞的现象,为了能够探测出更优的路径,应该增加可以选择路径的数目。通过这种动态的信息素调节机制可以有效地平衡收敛速度和避免早熟的现象。

在特殊情况下:

1) 当节点 i 中内容名称 c 对应的转发接口信息素值均相等时,即对于任意一个接口 j 信息素值 $\tau_{i,j}^c(x)$

$$\text{等于一个常数 } C, \text{ 则 } S_i^c(x) = \sqrt{\sum_{j=1}^{N_i^c} (E_i^c(x) - \tau_{i,j}^c(x))^2} = \sqrt{\sum_{j=1}^{N_i^c} (C - C)^2} = 0;$$

2) 当节点 i 中内容名称 c 对应的转发接口除了接口 j ,其他接口的值均为0时,则聚度值 $S_i^c(x) =$

$$\sqrt{\sum_{j=1}^{N_i^c} (E_i^c(x) - \tau_{i,j}^c(x))^2} = \sqrt{\sum_{j=1}^{N_i^c-1} (\frac{1}{N_i^c} - 0)^2 + (\frac{1}{N_i^c} - 1)^2} = \sqrt{1 - \frac{1}{N_i^c}}, \text{ 该值为聚度最大的可能值,记为 } S_{\max}^c(i);$$

3) 对于节点 i 中内容名称 c 对应只有一个转发接口时,以上公式仍然成立,只是接口的期望值 $E_i^c(x)$ 即为该接口信息素的值,而聚度值 $S_i^c(x)$ 则为0。

5.2 RED-ACO 优化算法

5.2.1 网络模型

设无向图 $G=(V,E)$ 表示一个网络, V 为节点集合, E 为网络节点之间的链路集合。RED-ACO中每个节点都含有一个信息素表 T_i ,任意一个内容名称 c 对应的信息素表为 T_i^c ,与内容名称 c 相关联的接口列表数为 N_i^c 。网络中可以通过多个指标反应网络的状态,如:网络负载、往返延时、网络抖动和网络拥塞等。为方便起见,只选取网络负载(load)和往返延时(delay)2个最主要的因素来衡量网络状态。则对于内容名称 c 接口列表中的任意一个接口 j ,与网络负载相关的信息素值为 $\tau_{i,j}^c(l)$,与往返延时相关的信息素值为 $\tau_{i,j}^c(d)$,分别将 $\tau_{i,j}^c(l)$ 和 $\tau_{i,j}^c(d)$ 归一化,使 $\sum_{j=1}^{N_i^c} \tau_{i,j}^c(x) = 1$,其中, c 为节点 i 中的任意一个内容名称, $x \in \{l, d\}$ 。

CCN节点中的FIB表用于将名称前缀和可以获取该名称前缀内容的接口列表关联起来。当兴趣分组到达CCN节点时,从FIB表中关联的接口列表选择一个接口转发出去。为了保证能够选择一个最佳的接口进行转发,需要考虑每个接口的多个网络因素,如:网络负载、可用带宽、链路拥塞状况和网络抖动等。为了能够在FIB表中快速选择一个最佳接口,RED-ACO中的每个网络节点都维护蚁群信息素表,如图2所示。

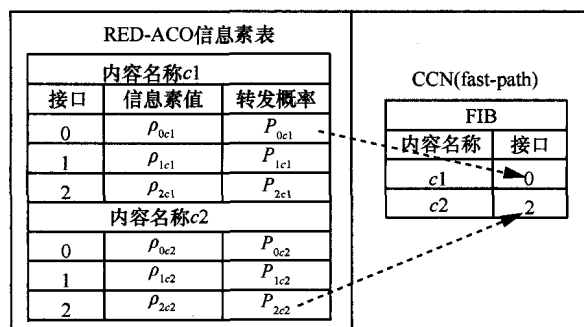


图2 RED-ACO 节点模型

信息素表中包含内容名称、关联的接口列表、相应的信息素值和对应的接口转发概率。对于每一个内容名称条目,通过运行蚁群算法计算每个接口信息素的值,然后根据信息素的值计算每个接口对应的转发概率,最后将转发概率最高的接口赋值到CCN的FIB表中。如图2所示,内容名称c1对应3个转发接口,其中接口0的转发概率值最大,故FIB表中对应内容名称c1的最佳转发接口为接口0;同样,假设内容名称c2的接口列表中接口2的转发概率值最大,则对应FIB表中内容名称c2的最佳转发接口为接口2。该节点模型相当于在CCN层面的基础上添加了RED-ACO层面,RED-ACO层面通过信息素表中的转发概率操作CCN中的FIB表,使FIB表中每个内容名称都只具有一个最佳的转发接口。所以,所有的兴趣分组请求都通过最佳接口转发从而得到一条最优路径。

5.2.2 算法描述

算法1 蚂蚁转发算法

S1: 请求节点生产兴趣蚂蚁(interesting ant):

While (local_time window of network node is starting)

//请求节点会根据FIB表中的内容名称随机选择一个内容名称生成兴趣蚂蚁;

generate IA(according to content name of FIB randomly);

S2: 转发兴趣分组:

1) if(IA's forwarding in requesting node)

push(local_time of self); //将自身的本地时间压栈;

forwarding(to all interfaces which matching with content name); //向与内容名称匹配的所有接口进行转发;

2) while(IA did not reach data node&&reach intermediate routing node)

//根据接口列表的每个接口对应的概率值随机选择接口转发;

forwarding(to interface which generated according to probability value);

push(forwarding time);

S3: 数据节点产生数据蚂蚁(data ant):

1) if(data node receive IA)

//数据节点产生一个数据蚂蚁;数据蚂蚁中包含兴趣蚂蚁中时间栈的一份拷贝和网络负载等网络状态信息。

generate DA (by data node);

back(to requesting node by original way)

while(DA reach intermediate routing node)

//计算往返延时,预示着网络路径的拥塞状况

Calculate Round-trip delay(time of receiving DA - forwarding time of self);

Collect network load by Ant Agent;

Updated strategy (of pheromones according to Round-trip delay and network load);

Modify pheromones(according to Updated strategy of pheromones);

Volatile operation(other interfaces);

2) else while(time window ended)

Volatile operation(all interfaces);

结合CCN自适应转发策略,如果兴趣蚂蚁经过某节点的一特定接口转发后,接收到网络上游节点发送的NACK兴趣分组,说明该接口对应的链路无数据蚂蚁,则快速结束本次探测,并对所经过的路由节点的相关接口信息素做挥发操作。另外,如果要选择的转发接口*i*已经达到最大的数据速率 L_i ,则不再选择该接口进行转发,并向网络下游节点发送NACK兴趣分组。

以上整个过程在网络的每个节点上运行,所以节点的信息表不但根据自身请求获得的数据蚂蚁进行更新,还根据其他节点经过的数据蚂蚁更新。正常的兴趣分组只根据蚁群算法形成的最佳接口进行转发,并且不对节点的信息素表操作。

算法2 基于分布均匀度蚁群算法

S1: {

计算整体信息素 $\tau_{i,j}^c$:

$$\tau_{i,j}^c = \alpha \tau_{i,j}^c(l) + (1 - \alpha) \tau_{i,j}^c(d)$$

其中, $\tau_{i,j}^c(l)$ 为负载信息素, $\tau_{i,j}^c(d)$ 为延时信息素, α 用于决定负载信息素 $\tau_{i,j}^c(l)$ 和延时信息素 $\tau_{i,j}^c(d)$ 对整体信息素 $\tau_{i,j}^c$ 的贡献权重。 α 的选择完全取决于网络的策略和需求, 并且可以根据每个内容名称进行设置。

}

S2: {

计算最多可进行转发的接口数 W_i^c :

$$W_i^c = \left\lfloor \frac{S_i^c}{S_{\max}^c} (N_i^c - 1) + 0.5 \right\rfloor + 1$$

其中, $S_{\max}^c(i)$ 表示节点 i 上内容名称 c 对应的接口列表聚度最大值。

}

S3: 将内容名称 c 相关联的转发接口列表的信息素值按降序排列, 将排序后的序号依次保存到 $\text{rank}[c][\cdot]$ 数组中。

S4: 确定信息素更新策略与更新信息素。

S5: 挥发策略与操作。

算例 1 在某网络通信中, 往返延时是最主要的考虑因素, 则 α 的值就会较小。节点 i 上内容名称 c 对应的接口列表聚度最大值记为 $S_{\max}^c(i)$, 最多可进行转发的接口数为 W_i^c , 则

$$W_i^c = \left\lfloor \frac{S_i^c}{S_{\max}^c} (N_i^c - 1) + 0.5 \right\rfloor + 1 \quad (10)$$

式(10)存在 2 种特殊情况。

1) 当信息素平均分配到各个接口上, 即聚度为 0 时, W_i^c 值为 1。这表示当信息素分配均匀时只能选择一个接口进行转发, 加快了求解的收敛速度。

2) 当信息素完全集中到某个特定接口上, 即聚度达到了最大值 $S_{\max}^c(i)$ 时, W_i^c 的值为 N_i^c 。这表示当信息素聚度达到最大时, 可以选择任何一个接口转发, 减小出现早熟现象的概率。

$\text{rank}[c][j]$ 元素的值为转发接口 j 的序号。设 $\xi_{i,j}^c$

为节点 i 上内容名称 c 对应的转发接口 j 的权重, q 为权重的公比, $q = W_i^c / N_i^c$, 则

$$\xi_{i,j}^c = \begin{cases} q^{\text{rank}[c][j]-1}, & \text{rank}[c][j] \leq W_i^c \\ 0, & \text{其他} \end{cases} \quad (11)$$

设 $P_{i,j}^c$ 为节点 i 上内容名称 c 相关联的转发接口 j 的转发概率, 则

$$P_{i,j}^c = \frac{\xi_{i,j}^c (\alpha \tau_{i,j}^c(l) + (1-\alpha) \tau_{i,j}^c(d))}{\sum_{j=1}^{N_i^c} \xi_{i,j}^c (\alpha \tau_{i,j}^c(l) + (1-\alpha) \tau_{i,j}^c(d))} \quad (12)$$

q 的取值范围为 $[1/N_i^c, 1]$, 接口列表的权重 $\xi_{i,j}^c$ 是以 q 为公比的等比数列。在算法初始运行时, 接口列表的聚度较小, W_i^c 和 q 的值均较小, 接口之间的权重相差较大; 使接口之间的转发概率相差较大, 可以加快求解的收敛速度。在特殊情况下, 接口之间的信息素值相同, 接口列表的聚度为 0, W_i^c 值为 1, 则第一个接口的转发概率为 1, 其他接口转发概率为 0。当接口列表的聚度较大时, 容易产生早熟的现象。此时, W_i^c 和 q 的值均较大, 接口之间的权重相差较小, 使接口之间的转发概率相差相对较小, 减小了出现早熟现象的概率。在特殊情况下, 信息素完全集中在某一特定接口上时, 可转发的接口数 W_i^c 等于接口列表的总数 N_i^c , 则接口的转发概率仅与接口的信息素值相关。

当节点 i 从接口 j 收到内容名称 c 对应的数据蚂蚁时, 算法通过获取的网络负载和计算出的往返延对接口 j 的信息素值进行更新。往返延时由接收数据蚂蚁的时间与时间栈中对应的转发时间的差得出。根据信息素的更新策略完成对相应接口信息素值的更新。信息素的更新策略是在当前值的基础上再加上信息素增量的值。信息素的增量与网络负载和往返延时成反比。另外, 接口列表的聚度越小, 则应该加强信息素的增加程度来加快收敛速度, 所以信息素的增量就越大; 而接口列表的聚度越大, 则应该减慢信息素的增加程度来避免早熟现象, 所以信息素的增量就越小。因此, 信息素的增量与接口列表的聚度成反函数关系。最后, 信息素增加的程度和当前值也成反函数关系, 应该减小比较大的信息素值增加的速度。综合上述所有因素, 信息素更新策略为

$$\tau_{i,j}^c(x) = \tau_{i,j}^c(x) + \Delta_x \quad (13)$$

其中, $\tau_{i,j}^c(x)$ 为信息素的值, Δ_x 为信息素的增量。

$$\Delta_x = -(x-1) \exp(x) (1-S_i^c) (1-\tau_{i,j}^c(x)) \quad (14)$$

对式(14)的解释如下: 首先, 节点 i 内容名称 c 的接口列表的聚度 S_i^c 的取值范围为 $\left[0, \sqrt{1-\frac{1}{N_i^c}}\right]$, $1-S_i^c$ 为正值且 S_i^c 与 Δ_x 成反比。然

后, $-(x-1)\exp(x)$ 中 x 代表网络负载和往返延时, 表示信息素增量 Δ_x 与 x 成反函数。

证明

$$\begin{aligned} f(x)' &= (-(x-1)e^{(x)})' \\ &= (1-x)'e^{(x)} + (1-x)e^{(x)'} \\ &= -e^{(x)} + (1-x)e^{(x)} \\ &= -xe^{(x)} \end{aligned}$$

x 的值为网络负载和往返延时分别进行归一化之后的数值, 即 $0 < x < 1$ 。当 x 大于 0 时, 导数小于零, 为减函数, 即 Δ_x 随着网络负载和往返延时的增大而减小。

在节点 i 通过接口 j 收到数据蚂蚁后, 其他与内容名称 c 关联的所有接口 j' 相应的信息素做挥发操作。设挥发系数为 ρ , 则接口挥发方程为

$$\tau_{i,j'}^c(x) = \tau_{i,j'}^c(x) - \rho \tau_{i,j'}^c(x) \quad (15)$$

如果节点最终没有收到数据蚂蚁, 则在时间窗口结束时刻对所有匹配的内容名称接口执行挥发操作。

6 仿真实验与结果分析

实验共分 2 部分进行, 首先验证转发节点模型。转发模型属于 0-1 整数规划问题, 有多种求解方法, 支持的工具有 Matlab 和 Lingo/Lindo 等。由于 Lingo10 具备灵活的输入输出及强大的编程能力, 在处理复杂的整数规划方面较 Matlab 更加灵活, 因此采用 Lingo10 进行模型求解。给出随机数据, 观测各种参数对决策结果的影响。其次验证在转发节点集合的基础上, RED-ACO 算法及其影响因素。

6.1 转发节点集合的确定验证

给出一个拥有 9 个节点组成的网络, 其中, 节点 $N_1 \sim N_8$ 为转出端口, 节点 N_0 为输入端口, N_0 与其他 8 个节点间的数据传输速度、连接状况因子以及转出端口的可用性如表 1 所示。

由于在服务质量函数基础上, 模型对传输时间

范围约束、网络负载函数进行了综合, 模型中引入了多个影响参数, 因此需要检测模型的有效性和这些参数对模型决策结果的影响程度。具体包括: z, r 对 t, λ 的影响; q 对 t, λ 的影响; π 对 f, λ 的影响。

6.1.1 z, r 对 t, λ 的影响

本实验主要检测服务质量函数的有效性, 以及其中各参数对决策结果的影响。权重系数 μ 的确定方法, μ 应该取这样的值: 在 π 取不同值时, 能够控制 f, λ 具有同级或相差 1~2 个数量级, 这样可以明显地区分出优化关注点, 即网络负载或节点资源, 通过实验观察, f, λ 一般相差 3 个数量级, 因此取 $\mu=1\ 000$ 。

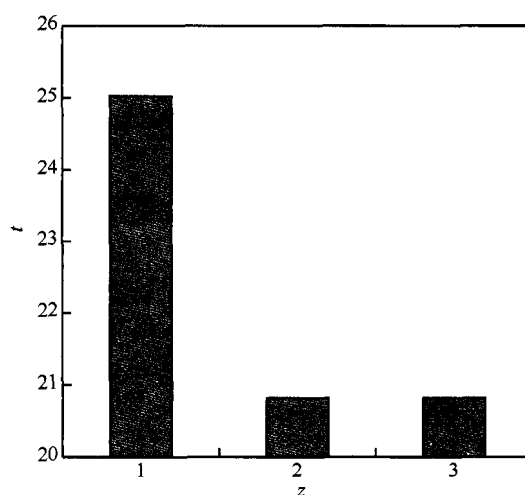


图3 $r=1$ 时 z 对 t 影响

令 $r=1, q=2\ 000$, 观测 z 对 t, λ 的影响。从图 3 和图 4 中看出, 当关键等级升高时, 决策结果产生了影响, 分别降低了传输时间, 提升了网络代价。但是由于受时间范围和节点个数约束, 当 $z=2, 3$ 时, 决策结果是相同的。

令 $z=2, q=2\ 000$, 观测 r 对 t, λ 的影响。从图 5 和图 6 中看出, 当服务质量提高时, 决策结果产生了影响, 分别降低了传输时间, 提升了网络代价。但是由于受时间范围和节点个数约束, 当 r 取某些值时, 决策结果是相同的。

表 1

各节点 $a_i, V_i, n_i b_i$

| 参数 | 节点 | | | | | | | |
|-----------|-------|-------|-------|-------|-------|-------|-------|-------|
| | N_1 | N_2 | N_3 | N_4 | N_5 | N_6 | N_7 | N_8 |
| a_i | 0.8 | 0.88 | 0.8 | 0.85 | 0.78 | 0.88 | 0.86 | 0.72 |
| V_i | 250 | 250 | 440 | 320 | 710 | 340 | 200 | 600 |
| $n_i b_i$ | 2 | 7 | 1 | 6 | 3 | 8 | 2 | 4 |

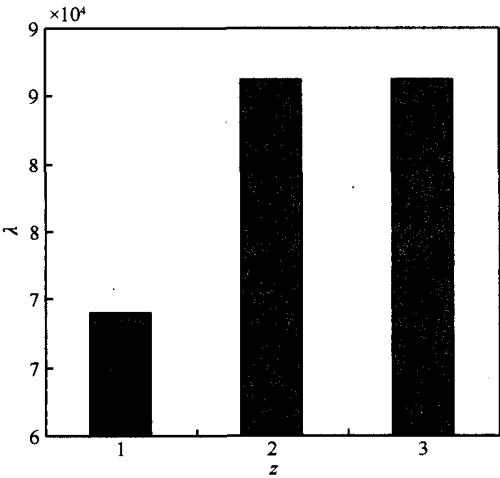


图 4 $r=1$ 时 z 对 λ 影响

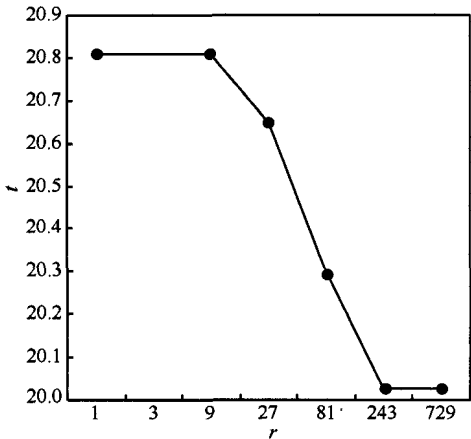


图 5 $z=2$ 时 r 对 t 影响

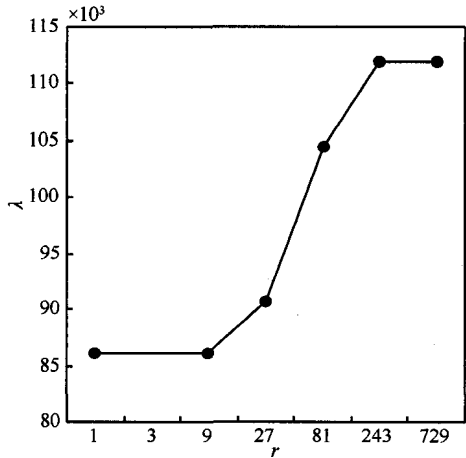


图 6 $z=2$ 时 r 对 λ 影响

从该实验可以看出，容忍度函数对模型的决策有直接的影响，并且参数 r, λ 对容忍度函数是有效的。

6.1.2 q 对 t, λ 的影响

令 $z=2, r=1, \mu=1\ 000$ 。观测 q 对 t, λ 的影响。从图

7 中看出，随着 q 的增加， t 有明显的降低。而由于 q 的增加使节点的选择具有更大空间，因此，从图 8 中可以看出， λ 随着 q 的增加呈现大幅下降趋势。

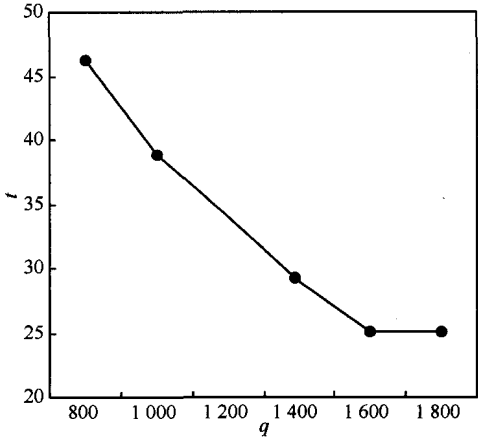


图 7 q 对 t 的影响

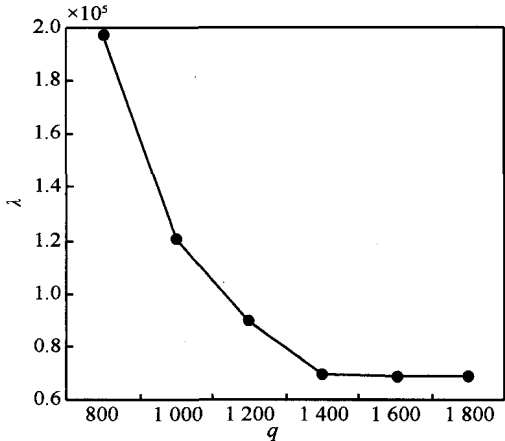


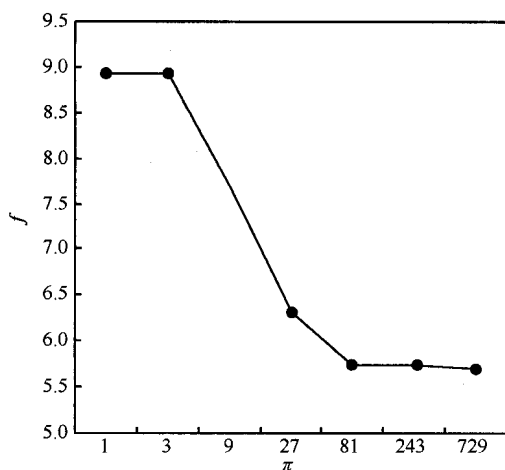
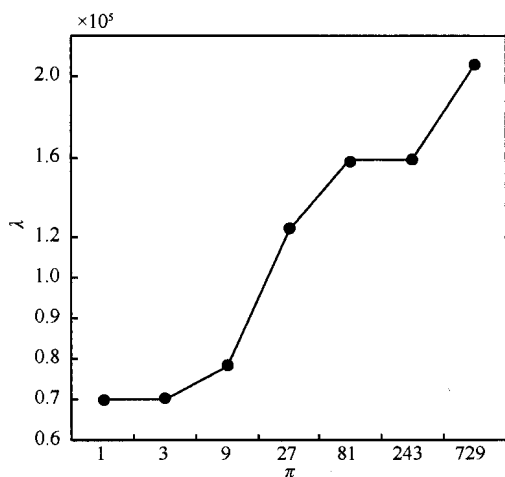
图 8 q 对 λ 的影响

6.1.3 π 对 f, λ 的影响

本实验主要测试权重因子 π 是否对优化目标中服务质量函数值 f 和网络负载 λ 起到均衡的作用。令 $r=1, z=2, q=2\ 000, \mu=1\ 000, \pi=\{3^0, 3^1, 3^2, 3^3, 3^4, 3^5, 3^6\}$ ，观测 π 取不同的值对 f, λ 的影响。

从图 9 和图 10 中看出，随着 π 的增加，相应的 f 呈现大幅下降趋势，而 λ 则呈现大幅上升趋势。权重因子确实起到了均衡两者的作用。

通过引入服务质量函数，将服务请求对传输时间的服务质量度进行了量化，模型不只关注于网络负载和传输时间，而是可以根据服务对时间需求程度和当前的网络状态进行动态决策。模型中的权重因子对服务的度量取值和网络负载起到了较好的调节作用，各种参数 r, z, q, π 对模型的决策起到了有效的调节作用，模型具有较好的适用性和可行性。

图9 π 对 f 的影响图10 π 对 λ 的影响

6.2 RED-ACO 算法验证实验

6.2.1 实验参数设置

实验基于 NS-3 的 ndnSIM 仿真平台和 CCNx 协议,模拟实验网络拓扑图通过 GT-ITM 工具生成。具体参数如表 2 所示。

| 参数名称 | 参数 |
|-------|--------------|
| 网络节点数 | 50 |
| 相连概率 | 20% |
| 带宽范围 | 10~20 Mbit/s |
| 时延范围 | 5~10 ms |

如表 2 所示,每 2 个节点相连的概率为 0.2,每条网络链路的带宽和网络时延随机设置,带宽值为 10~20 Mbit/s,网络时延值为 5~10 ms。随机选择 10 个网络节点作为数据源节点,选择 10 个网络节

点作为数据请求节点,其他节点均为路由节点。路由节点可以产生兴趣蚂蚁和数据蚂蚁,分别运行 SoCCeR 优化算法和 RED-ACO 优化算法。优化算法模型中,网络负载权重系数 α 为 0.8,信息素挥发系数 ρ 为 0.2,SoCCeR 优化算法随机探测概率为 0.2。

由于 RED-ACO 算法和 SoCCeR 算法都将往返延时和网络负载作为信息素更新的主要因素,模拟实验分别对网络的平均延时和负载率方差进行了统计和分析。

6.2.2 平均延时的分析

网络平均延时反映所有请求节点在发送兴趣分组后到接收到相应数据分组的网络延时的均值,是一个重要的性能指标。图 11 为 SoCCeR 算法和 RED-ACO 算法的平均延时对比。因为 SoCCeR 算法和 RED-ACO 算法在最初运行时,不能直接探测出所有网络节点的最优通信链路;所以两者在最初阶段平均延时都相对较高。但是,RED-ACO 算法平均延时的下降速度要快于 SoCCeR 算法的下降速度,更早到达一个相对较稳定的水平。因为最初阶段转发接口列表的聚度值都相对较小,RED-ACO 算法收敛速度更快。当算法运行到 25 周期时,实验增加了某些链路的带宽,RED-ACO 算法平均延时下降速度略快于 SoCCeR 算法的速度。在第 35 周期时,减小某些链路的延时,RED-ACO 算法在之后若干周期的平均延时略低于 SoCCeR 算法的平均延时。第 45 周期左右时刻,2 种算法的平均延时都趋于稳定在一定范围内。

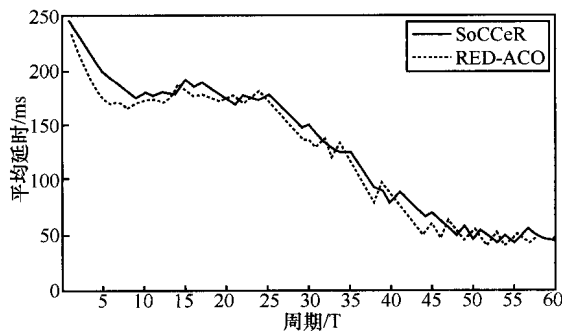


图11 平均延时对比

6.2.3 负载率方差的分析

负载率指一个网络节点实际承受的网络负载和该节点最大网络负载的比值。负载率的方差反映网络节点中负载率的差别程度;若负载率方差越小,则表示节点间负载越均衡。SoCCeR 算法和

RED-ACO 算法的负载率方差对比如图 12 所示。在模拟实验最初运行阶段, 网络节点之间的负载率差别较大; 所以, SoCCeR 算法和 RED-ACO 算法的负载率方差在初始阶段均相对较大。在最初阶段, 方差值下降的过程中, RED-ACO 算法的速度略快于 SoCCeR 算法的速度, 并且 RED-ACO 方差的值小于 SoCCeR 方差的值。因为最初阶段蚁群仍相对分散, 聚度值较小, RED-ACO 的收敛速度较快。在第 25 周期和第 35 周期, 分别提高了某些链路的带宽和减小某些链路的延时, 造成某些节点的负载发生变化, 从而出现了负载率方差小幅度增加的现象。在第 45 周期左右, RED-ACO 算法的方差值和 SoCCeR 算法的方差值均相对较小, 并一直保持在一个比较稳定的水平。根据以上对比分析得知, RED-ACO 算法和 SoCCeR 算法对网络负载的均衡都具有良好的性能, RED-ACO 算法在最初阶段更快地达到一个相对稳定的水平, 收敛速度较快。

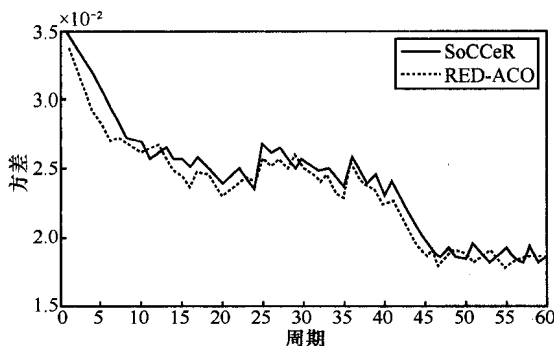


图 12 负载率方差对比

7 结束语

内容中心网络以命名数据为中心解决了现有网络存在的不匹配问题。路由与转发策略是网络的核心机制之一, 路由算法的性能对网络的各个参数状态具有重大的影响。蚁群算法是一种生物启发式算法, 在网络路由优化上有广泛的应用。本文针对蚁群算法自身存在的收敛速度慢和易产生停滞现象的问题, 结合内容中心网络本身体系架构具有的优势, 提出了转发接口集合的选择模型, 通过提出服务质量的概念来确定转发接口的集合, 在此基础上设计了一种基于分布均匀度和转发策略相结合的 CCN 蚁群路由优化算法 RED-ACO。仿真结果表明, 该算法在内容请求时延和降低网络负载方面性

能优于 SoCCeR 路由优化算法。

参考文献:

- [1] LAGUTIN D, VISALA K, TARKOMA S. Publish/subscribe for Internet: PSIRP Perspective[M]. Amsterdam: IOS Press. 2010. 75-84.
- [2] TARKOMA S, AIN M, VISALA K. The Publish/Subscribe Internet Routing Paradigm (PSIRP): Designing the Future Internet Architecture[M]. Amsterdam: IOS Press. 2009. 102-111.
- [3] JACOBSON V, SMETTERS D K, THORNTON J D. Networking named content[A]. Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies[C]. New York, NY, USA, 2009. 1-12.
- [4] GREGLER H, GORALSKI W. Router Architecture The Complete IS-IS Routing Protocol[M]. Springer London Ltd, 2005. 11-34.
- [5] PELISSIER B, RHODES W, SAYLOR WG, *et al.* TRIAD drug treatment evaluation project[J]. Fed Probation, 2000, 65(3): 3-7.
- [6] KOPONENT T, CHAWLA M, CHUN B, *et al.* A data-oriented (and beyond) network architecture[J]. ACM SIGCOMM Computer Communication Review, 2007, 37(4): 181-192.
- [7] KIM J, JANG M W, LEE B J B, *et al.* Content centric network-based virtual private community[A]. Consumer Electronics (ICCE) 2011 IEEE International Conference on[C]. Las Vegas, NV, USA, 2011. 843-844.
- [8] CEBROWSKI A K, GARSTKA J J. Network-centric warfare: its origin and future[J]. U S Naval Institute Proceedings, 1998, 124(1): 8-35.
- [9] LEINER B M, CERF V G, CLARK D D, *et al.* A brief history of the Internet[J]. ACM SIGCOMM Computer Communication Review, 2009, 39(5): 22-31.
- [10] United state NSF future Internet architecture project[EB/OL]. <http://www.nets-fia.net/>.
- [11] SUN X X, WANG X W, LI J, *et al.* A novel routing scheme for ICN[J]. Applied Mechanics and Materials, 2014, 543: 3555-3560.
- [12] ZENG X, GAO Z H. Rank-based routing strategy for named data network[J]. Applied Mechanics and Materials, 2014, 543: 3320-3323.
- [13] WANG Y, ZU Y, ZHANG T, *et al.* Wire speed name lookup: a gpu-based approach[A]. Proceedings of the 10th USENIX Conference on Networked Systems Design and Implementation[C]. Berkeley, CA, USA, 2013. 199-212.
- [14] 叶润生, 徐明伟. 命名数据网络中的邻居缓存路由策略[J]. 计算机科学与探索. 2012, 6(7): 593-601.
YE R S, XU M W. Neighbor cache explore routing strategy in named data network[J]. Journal of Frontiers of Computer Science and Technology, 2012, 6(7): 593-601.
- [15] LI C, OKAMURA K, LIU W. Ant colony based forwarding method for content-centric networking[A]. Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference on[C]. Barcelona, Spain, 2013. 306-311.
- [16] ARIANFAR S, NIKANDER P, OTT J O R. On content-centric router design and implications[A]. ReARCH'10[C]. Philadelphia, PA, USA, 2010.
- [17] SHANBHAG S, SCHWAN N, RIMAC I, *et al.* SoCCeR: services

over content-centric routing[A]. Proceedings of the 2011 ACM SIGCOMM Workshop on Information-Centric Networking[C]. Toronto, Canada, 2011. 62-67.

- [18] SUN L, SONG F, YANG D, *et al.* DHR-CCN, distributed hierarchical routing for content centric network[J]. Journal of Internet Services and Information Security (JISIS), 2013, 3(1/2): 71-82.
- [19] EYMANN J, TIMM-GIEL A. Multipath Transmission in Content Centric Networking Using a Probabilistic Ant-Routing Mechanism, in Mobile Networks and Management[M]. Springer, 2013. 45-56.
- [20] LI C M, LIU W J, OKAMURA K. A greedy ant colony forwarding algorithm for named data networking[J]. Proceedings of the Asia-Pacific Advanced Network, 2013, 34: 17-26.
- [21] KHAN A Z, BAQAI S, DOGAR F R. QoS aware path selection in content centric networks[A]. 2012 IEEE International Conference on ICC[C]. Ottawa, ON, Canada, 2012. 2645-2649.
- [22] 曲大鹏, 王兴伟, 黄敏. 移动对等网络中的感知蚁群路由算法[J]. 计算机学报, 2013, 36(7): 1456-1464.
QU D P, WANG X W, HUANG M. An aware ant routing algorithm in mobile peer-to-peer networks[J]. Chinese Journal of Computers, 2013, 36(7): 1456-1464.
- [23] AL-ALI R, RANA O, WALKER D W, *et al.* G-QoS: grid service discovery using QoS properties[J]. Chinese Journal of Computers, Computing and Informatics. 2012, 21(4): 363-382.
- [24] HESTNES B O R, BROOKS P, AABY C, *et al.* Quality of experience in real-time person-person communication-user based QoS expressed in technical network QoS terms[A]. Proceedings of the 19th International Symposium on Human Factors in Telecommunication[C]. Berlin, Germany, 2003. 3-10.
- [25] TIPSUWAN Y, CHOW M. Network-based controller adaptation based on QoS negotiation and deterioration[A]. Industrial Electronics Society, 2001(IECON '01)[C]. Denver, CO, USA, 2001. 1794-1799.
- [26] GEORGIADIS L, GUERIN R, PERIS V, *et al.* Efficient network QoS provisioning based on per node traffic shaping[J]. IEEE/ACM Transactions on Networking (TON), 1996, 4(4): 482-501.

作者简介:



张国印(1962-), 男, 山东黄县人, 哈尔滨工程大学教授、博士生导师, 主要研究方向为网络与信息安全、嵌入式系统等。



唐滨(1985-), 男, 黑龙江牡丹江人, 哈尔滨工程大学博士生, 主要研究方向为网络与信息安全、下一代网络系统结构。



孙建国(1981-), 男, 黑龙江巴彦人, 哈尔滨工程大学副教授、硕士生导师, 主要研究方向为信息安全、嵌入式系统设计。



李佳楠(1994-), 女, 黑龙江哈尔滨人, 主要研究方向为网络信息安全、下一代网络体系结构。