

基于效用函数簇的效用类服务分层调度模型

宋亚楠^{1,2}, 仲 茜², 胡成臣³, 刘 斌¹

(1. 清华大学计算机科学与技术系, 北京 100084; 2. 72241 部队, 山东济南 250029;

3. 西安交通大学计算机科学与技术系, 陕西西安 710049)

摘 要: 针对基于效用的单层调度模型对网络流划分粒度较粗, 难以实现服务间效用公平和总效用最大化的问题, 提出了 U-S 两层调度模型, 设计了高效求解算法 USMS, 为提高求解精度, 进一步提出了 USIM 算法. 仿真实验结果显示: USMS 和 USIM 算法具有良好的求解有效性和效率; 与单层模型相比, U-S 模型具有更高的总效用值.

关键词: 效用; 效用类; 服务; 调度

中图分类号: TP393.05

文献标识码: A

文章编号: 0372-2112 (2012) 02-0247-07

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2012.02.007

Utility Function Group Based Utility Class and Service Hierarchical Scheduling Model

SONG Ya-nan^{1,2}, ZHONG Qian², HU Cheng-chen³, LIU Bin¹

(1. Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China;

2. 72241 Unit, Jinan, Shandong 250029, China;

3. Department of Computer Science and Technology, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China)

Abstract: Due to the shortcoming of coarse partitioning granularity, utility-based single layer scheduling model fails to achieve good fairness among services and to maximize the whole utility. A two-layer hierarchical scheduling model named U-S as well as the corresponding algorithm USMS to solve the model is proposed. In order to improve the accuracy of the solution, we further design an improved algorithm named USIM. The evaluation experiments show that both USMS and USIM can achieve high solving effectiveness and efficiency and also have higher average whole utility than the single-layer model.

Key words: utility; utility class; service; scheduling

1 引言

为不同应用提供不同服务质量保证已经成为网络资源调度的研究热点. 随着 Internet 的发展, 网络中出现了越来越多的新服务, 它们对资源的要求各不相同. 例如, 高清电视需要严格的带宽保证, 而电子邮件则没有对带宽提出严格要求. 鉴于传统基于带宽共享的资源调度策略在满足用户对网络服务质量要求方面存在不足, Shenker^[1]引入了效用 (Utility) 的概念, 他将网络资源映射到网络服务的性能表现, 包括除了一般服务质量参数外的更多度量, 如页面显示或文件传送速度、音频质量、图像质量、视频连续度等.

基于效用的资源分配研究已在不同网络环境^[2~5]中展开, 出现了许多研究成果. 如 Cao 等^[6]提出的基于

效用最大最小公平的调度策略; Harks^[7]等提出的适用范围更广泛且对效用函数无严格要求的效用比例公平准则等. 特别是, Liu^[8]等对下一代互联网服务的效用函数进行了深入研究, 按照流 (五元组 < 源 IP 地址, 目的 IP 地址, 源端口, 目的端口, 协议域 >) 的效用函数将其划分为五类, 即 VoIP、IPTV、HTTP、TCP 和 UDP (本文中每一类称为一个效用类), 并提出了基于效用的调度模型 (UBSFTP: Utility Based Scheduling For Trip-Play), 目标是实现网络总效用最大化. 上述模型建立在同一效用类中所有流都具有相同效用函数和实际带宽分配这一假设上. 该假设简化了调度模型, 降低了求解的复杂度, 因此被许多调度模型采用. 然而, 这种假设也存在着由于不区分同一效用类中流的所属服务, 导致无法实现服务间真正公平, 进而无法实现基于效用公平的总效用最大

化的问题.为此 Satya 等^[9]在文献[8]的基础上,对流进行更为详细地分析,取消了同一效用类下的流具有相同带宽的假设,初步建立了效用类-流分层调度模型,但依然没有对效用类中流所属的服务进行区分,每一效用类下的流仍具有完全相同的效用函数,文献[8]中所存在的问题并未得到有效地解决.

为此,本文提出了基于效用的两层调度模型:效用类-服务 U-S (Utility Class and Service) 模型,图 1 为模型示意图.在使用效用类对流进行划分的基础上,进一步按网络服务对其作了区分.在合理反映路由器中流调度过程的同时,实现了基于效用公平的模型总效用最大化的目标.针对 U-S 模型,本文作了深入地研究,主要贡献有:(1)提出效用函数簇的概念,以描述同一效用类中不同网络服务的效用函数;(2)利用效用类中各服务间实际带宽分配的关系对 U-S 模型进行简化;(3)设计了 USMS(U-S Model Solution) 模型求解算法和进一步提高求解精度的 USIM(U-S Iterative Modification)算法.为验证上述算法的有效性,将其与经典优化模型求解软件 Lingo9.0 中的相应算法作了对比,结果表明:(1)文中算法的解与 Lingo9.0 中算法的解相近.USMS 算法的解与 Lingo9.0 的解平均仅相差 0.4%,USIM 算法可以提高 USMS 算法的求解精度(平均高 1.1%);(2)两种算法的求解时间远远小于 Lingo9.0 中算法的求解时间,分别只有 Lingo9.0 中算法求解时间的 1/10000 和 1/5000;(3)与 UBSFTP^[8]模型相比,在拥塞度 * 1 到 10 下,U-S 模型均具有更高的总效用值(平均高 22.7%,最大高 50.58%).

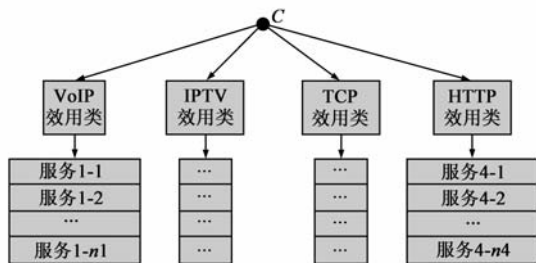


图1 效用类-服务(U-S)两层调度模型

2 相关概念

定义 1:效用 在网络中,效用通常表示为用户使用网络时对服务质量的满意程度.

定义 2:效用函数 效用的量化表示形式,不失一般性,本文中的效用函数为带宽的函数,记作 $U(r_i)$, r_i 为服务 i 分配到的带宽.

定义 3:效用函数簇 由属于同一效用类下不同服务的效用函数组成,通过参数化效用函数中的某些常数来实现.

3 效用类的效用函数簇

3.1 服务的最大最小带宽需求

本文中,服务的最大带宽需求是使效用函数值达到最大(通常为 1)的最小带宽;最小带宽需求是网络服务达到可用时所需要分配到的最小带宽,实际带宽若小于最小带宽则效用值为 0.目前基于效用的调度模型研究,如文献[8],通常假设同一效用类下的服务都具有相同的最大最小带宽需求,分别记为 $b_{\max i}$ 和 $b_{\min i}$ (i 表示第 i 个效用类).由于不同服务具有不同的带宽需求,上述假设并不合理.因此在 U-S 模型中,针对具体服务设置最大最小带宽需求,分别记为 $b_{\max i,j}$ 和 $b_{\min i,j}$, i 表示第 i 个效用类, j 表示第 j 个服务,描述如下:

$$b_{\max i,j} = \min(a_{i,j}, b_{\max i})$$

$$b_{\min i,j} = \begin{cases} 0, & (a_{i,j} < b_{\min i}) \\ b_{\min i}, & (a_{i,j} \geq b_{\min i}) \end{cases}$$

其中, $a_{i,j}$ 为服务的带宽需求.设第 i 个效用类的总带宽需求用 A_i 表示, $A_i = \sum_{j=1}^{n_i} a_{i,j}$. n_i 为第 i 个效用类中的服务个数,设第 i 个效用类的最大最小带宽需求分别用 $B_{\max i}$ 和 $B_{\min i}$ 来表示, $B_{\max i} = \sum_{j=1}^{n_i} b_{\max i,j}$, $B_{\min i} = \sum_{j=1}^{n_i} b_{\min i,j}$.

3.2 效用类的效用函数簇

参考文献[8],考虑到 IPTV 和 UDP 具有相似的效用函数曲线,本文将服务在效用类层划分为四类,即 VoIP、IPTV、TCP 和 HTTP,下面给出它们的效用函数簇.在本文中,使用 $u_{i,j}$ ($i \in \{1,2,3,4\}$) 表示第 i 个效用类的第 j 个服务的效用函数, $x_{i,j}$ 表示第 i 个效用类的第 j 个服务的带宽.

(1) VoIP 效用类

VoIP 效用类的效用函数簇如图 2(a)所示,表达式为:

$$u_{1,j}(a_{1,j}, x_{1,j}) = (\text{sgn}(x_{1,j} - b_{\min 1,j}) + 1)/2 \quad (1)$$

$$b_{\min 1,j} \leq x_{1,j} \leq b_{\max 1,j} \quad (1 \leq j \leq n_1)$$

令 $b_{\max 1} = b_{\min 1} = 64\text{Kbps}$, 64Kbps 是大多数有线电话通讯的标准解码率.

(2) IPTV 效用类

IPTV 效用类的效用函数如图 2(b)所示,其表达式为:

$$u_{2,j}(a_{2,j}, x_{2,j}) = 1/(1 + (1/\epsilon - 1)e^{-r_{2,j} \cdot x_{2,j}})$$

$$r_{2,j} = 2\ln(1/\epsilon - 1)/b_{\max 2,j} \quad (2)$$

$$b_{\min 2,j} \leq x_{2,j} \leq b_{\max 2,j} \quad (1 \leq j \leq n_2)$$

ϵ 是当服务分配到的带宽为 $b_{\min 2}$ 时所获得的效用

* 拥塞度为流的总带宽需求与链路输出总容量的比值.

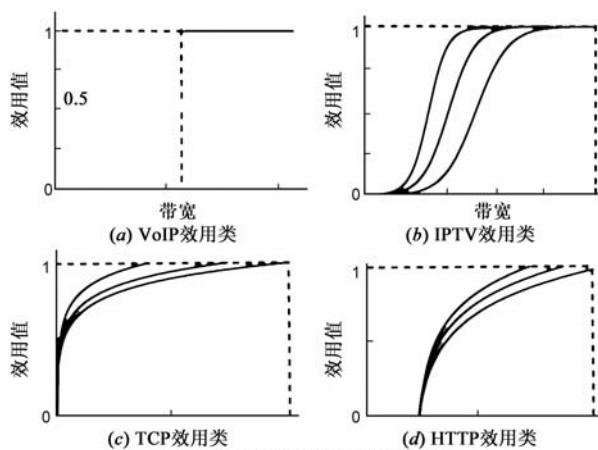


图2 效用类的效用函数簇曲线

值.按照最新的 MPEG-2 和 MPEG-4/H.264 解码率,设 $b_{\min 2} = 100\text{Kbps}$, $b_{\max 2} = 10\text{Mbps}$, $\epsilon = 0.001$.

(3) TCP 效用类

TCP 效用类的效用函数簇曲线如图 2(c) 所示,表达式为:

$$u_{3,j}(a_{3,j}, x_{3,j}) = \log(x_{3,j} + 1) / \log(b_{\max 3,j} + 1) \quad (3)$$

$$b_{\min 3,j} \leq x_{3,j} \leq b_{\max 3,j} \quad (1 \leq j \leq n_3)$$

令 $b_{\min 3} = 0$, $b_{\max 3} = 10\text{Mbps}$.

(4) HTTP 效用类

HTTP 效用类的效用函数簇如图 2(d) 所示,表达式为:

$$u_{4,j}(a_{4,j}, x_{4,j}) = ((\text{sgn}(x_{4,j} - b_{\min 4,j}) + 1) / 2) \times (\log(x_{4,j} / b_{\min 4,j}) / \log(b_{\max 4,j} / b_{\min 4,j})) \quad (4)$$

$$b_{\min 4,j} \leq x_{4,j} \leq b_{\max 4,j} \quad (1 \leq j \leq n_4)$$

令 $b_{\min 4} = 24\text{Kbps}$ 和 $b_{\max 4} = 10\text{Mbps}$.

4 U-S 模型及其简化

4.1 U-S 模型

U-S 模型分为效用类和服务两层,其目标是在满足同一效用类下各服务效用公平的同时实现总效用最大化.令 C 为输出链路的总容量, X_i 为效用类层第 i 个效用类的带宽分配, U 为 U-S 模型的总效用, U_i 为第 i 个效用类的总效用,参照[8]分别设 $V_1 = 1.0$, $V_2 = 9.0$, $V_3 = 1.0$ 和 $V_4 = 1.5$, U-S 模型表示为:

$$\max U = V_1 U_1 + V_2 U_2 + V_3 U_3 + V_4 U_4 \quad (5)$$

$$\text{St. } U_i = \sum_{j=1}^{n_i} u_{i,j} \quad (1 \leq i \leq 4) \quad (6)$$

$$u_{i,1} = u_{i,2} = \dots = u_{i,n_i} \quad (1 \leq i \leq 4) \quad (7)$$

$$\sum_{i=1}^4 \sum_{j=1}^{n_i} x_{i,j} \leq C \quad (8)$$

$$b_{\min i,j} \leq x_{i,j} \leq b_{\max i,j} \quad (1 \leq i \leq 4, 1 \leq j \leq n_i)$$

以上模型是一个非线性最优化模型,为了简化模

型的求解,需要对其进行化简.

4.2 效用类下服务间的实际带宽分配关系

利用式(7)可以推导出同一效用类中各服务间带宽的关系,减少未知数的个数,从而降低模型求解的复杂度.

(1) IPTV 效用类

$$\begin{aligned} \text{由 } 1/(1 + (1/\epsilon - 1)e^{((-2\ln(1/\epsilon - 1) \cdot x_{2,j})/b_{\max 2,j})}) \\ = 1/(1 + (1/\epsilon - 1)e^{((-2\ln(1/\epsilon - 1) \cdot x_{2,1})/b_{\max 2,1})}) \end{aligned}$$

$$\text{得 } x_{2,j} = x_{2,1} / b_{\max 2,1} \cdot b_{\max 2,j} \quad (2 \leq j \leq n_2) \quad (9)$$

同理得其它效用类服务间的带宽关系.

(2) TCP 效用类

$$x_{3,j} = (x_{3,1} + 1)^{\log(b_{\max 3,j} + 1) / \log(b_{\max 3,1} + 1)} - 1 \quad (2 \leq j \leq n_3) \quad (10)$$

(3) HTTP 效用类

当各服务带宽分配满足各自的最小带宽需求时有:

$$x_{4,j} = b_{\min 4,j} (b_{\max 4,j} / b_{\min 4,j})^{\log(x_{4,1} / b_{\min 4,1}) / \log(b_{\max 4,1} / b_{\min 4,1})} \quad (11)$$

4.3 模型简化

利用式(6)和(7),式(5)中模型等价于:

$$\begin{aligned} \max U = V_1 n_1 u_{1,1} + V_2 n_2 u_{2,1} \\ + V_3 n_3 u_{3,1} + V_4 n_4 u_{4,1} \end{aligned} \quad (12)$$

$$\text{St. } \sum_{i=1}^4 \sum_{j=1}^{n_i} x_{i,j} \leq C \quad (13)$$

$$b_{\min i,j} \leq x_{i,j} \leq b_{\max i,j} \quad (1 \leq i \leq 4, 1 \leq j \leq n_i)$$

由于 VoIP 中服务的最大最小带宽需求都是 64Kbps,带宽需求小,首先满足它们不会对其它效用类的效用值造成较大影响,同时又能获得较大的效用值,因此首先满足这类服务的带宽需求,即 $x_{1,j} = b_{\max 1}$,式(13)转化为:

$$\sum_{j=1}^{n_2} x_{2,j} + \sum_{j=1}^{n_3} x_{3,j} + \sum_{j=1}^{n_4} x_{4,j} \leq C - n_1 \cdot b_{\max 1} \quad (14)$$

由式(9)~(11),将式(14)改写为:

$$\begin{aligned} & \left(\sum_{j=1}^{n_2} b_{\max 2,j} \right) / b_{\max 2,1} \cdot x_{2,1} \\ & + \sum_{j=1}^{n_3} (x_{3,1} + 1)^{\log(b_{\max 3,j} + 1) / \log(b_{\max 3,1} + 1)} \\ & + \sum_{j=1}^{n_4} \left(b_{\min 4,j} (x_{4,1} / b_{\min 4,1})^{\log(b_{\max 4,j} / b_{\min 4,j}) / \log(b_{\max 4,1} / b_{\min 4,1})} \right) \\ & \leq C - n_1 b_{\max 1} + n_3 \end{aligned} \quad (15)$$

令 $C_1 = C - n_1 b_{\max 1} + n_3$

$$m = \left(\sum_{j=1}^{n_2} b_{\max 2,j} \right) / b_{\max 2,1}$$

$$p_j = \log(b_{\max 3,j} + 1) / \log(b_{\max 3,1} + 1)$$

$$q_j = \log(b_{\max 4,j} / b_{\min 4,j}) / \log(b_{\max 4,1} / b_{\min 4,1})$$

则式(15)可以写成:

$$m \cdot x_{2,1} + \sum_{j=1}^{n_3} (x_{3,1} + 1)^{p_j} + \sum_{j=1}^{n_4} (b_{\min 4,j} (x_{4,1}/b_{\min 4,1})^{q_j}) \leq C_1 \quad (16)$$

上述模型主要针对网络处于拥塞状态下的资源调度策略,因为当总带宽需求小于等于链路容量时,网络能够满足每一个服务的带宽需求,故它们的实际带宽分配等于带宽需求,即 $x_{i,j} = a_{i,j}$, 每一个服务的效用均为 1. 当网络发生拥塞时,所有服务的总带宽需求大于链路的总容量,即 $A > C$. 由于效用函数都是非减函数,带宽分配多,效用值就大,因此 U-S 模型获得最大效用值时,式(16)取等号. 即:

$$m \cdot x_{2,1} + \sum_{j=1}^{n_3} (x_{3,1} + 1)^{p_j} + \sum_{j=1}^{n_4} (b_{\min 4,j} (x_{4,1}/b_{\min 4,1})^{q_j}) = C_1 \quad (17)$$

于是

$$x_{2,1} = (C_1 - \sum_{j=1}^{n_3} (x_{3,1} + 1)^{p_j} - \sum_{j=1}^{n_4} (b_{\min 4,j} (x_{4,1}/b_{\min 4,1})^{q_j})) / m$$

根据以上分析,式(5)中模型转化为:

$$\text{Max } U = V_1 n_1 u_{2,1} + V_2 n_2 u_{2,1} + V_3 n_3 u_{3,1} + V_4 n_4 u_{4,1} \quad (18)$$

St.

$$x_{2,1} = (C_1 - \sum_{j=1}^{n_3} (x_{3,1} + 1)^{p_j} - \sum_{j=1}^{n_4} (b_{\min 4,j} (x_{4,1}/b_{\min 4,1})^{q_j})) / m \quad (19)$$

5 模型求解算法

虽然第 4 节对模型进行了化简,但由于模型仍为非线性优化模型,直接求解难度大,因此需要设计相应的近似求解算法.

5.1 求解算法

算法 1 模型求解算法 USMS

输入: $C, A, V_j, n_j (j \in [1, 4]), b_{\max i,j} (j \in [1, n_i], i \in [1, 4])$

输出: 最优解 U_{\max} 和对应的 $x_{2,1}, x_{3,1}, x_{4,1}$.

1. 计算 $C_1, m, p_j (j \in [1, n_3]), q_j (j \in [1, n_4])$.
2. Δ_3, Δ_4 为初始搜索步长, 当前最大效用值 U_{\max} 为 0.
3. 令 $x_{3,1}$ 的初值为 0.
4. 令 $x_{4,1}$ 的初值为 0.
5. 利用式(19)计算 $x_{2,1}$.
6. 如果 $x_{2,1} \geq 0$ 利用式(2)~(4)和(12)计算当前的 U , 否则转 9.
7. 如果 $U > U_{\max}$, $U_{\max} = U$ 存储当前 $x_{2,1}, x_{3,1}$ 和 $x_{4,1}$.
8. $x_{4,1} = x_{4,1} + \Delta_4$, 如果 $x_{4,1} \leq b_{\max 4,1}$ 转 5.
9. 如果 4~8 步的循环中 $x_{2,1}$ 均小于 0 转 11.
10. $x_{3,1} = x_{3,1} + \Delta_3$, 如果 $x_{3,1} \leq b_{\max 3,1}$ 转 4.
11. 返回 U_{\max} 和此时的 $x_{2,1}, x_{3,1}, x_{4,1}$.

由算法 1 可见, USMS 算法分别沿着 $x_{3,1}$ 和 $x_{4,1}$ 增加

的方向以步长 Δ_3 和 Δ_4 进行搜索, 对于每个 $x_{3,1}$ 和 $x_{4,1}$ 通过式(19)计算 $x_{2,1}$, 然后利用式(2)、(3)、(4)和(12)计算当前的效用值 U , 最后将搜索到的最大的 U 作为模型的最优解. 算法第 6 步中当 $x_{2,1} < 0$ 时, 对于比当前 $x_{4,1}$ 大的值必然会使 $x_{2,1} < 0$, 因此立即结束本次针对 $x_{4,1}$ 的搜索. 同样, 算法第 9 步表明对于当前 $x_{3,1} \geq 0$, 无论 $x_{4,1}$ 多小, 都无法使 $x_{2,1} \geq 0$, 因此无需对 $x_{3,1}$ 进行进一步搜索.

5.2 算法的复杂度和误差分析

(1) 复杂度分析

算法总共需要 $(b_{\max 3,1}/\Delta_3) \times (b_{\max 4,1}/\Delta_4)$ 趟迭代, 每一趟时间复杂度为 $O(n)$, $n = \max(n_3, n_4)$, 算法总的时间复杂度为 $O(n \times (b_{\max 3,1}/\Delta_3) \times (b_{\max 4,1}/\Delta_4))$.

(2) 误差分析

定理 1 设模型取得最优解 U_{\max}^0 时 $x_{2,1}, x_{3,1}, x_{4,1}$ 的值为 $x_{2,1}^0, x_{3,1}^0, x_{4,1}^0$, 则 USMS 算法迭代过程中存在模型的可行解 $(x'_{2,1}, x'_{3,1}, x'_{4,1})$ 使得 $x'_{3,1} \leq x_{3,1}^0 \leq x'_{3,1} + \Delta_3$ 且 $x'_{4,1} \leq x_{4,1}^0 \leq x'_{4,1} + \Delta_4 (\Delta_3, \Delta_4 \geq 0)$.

证明 由 USMS 算法的迭代过程可知, 在执行第 5 步时必存在 $x'_{3,1}, x'_{4,1}$ 使得 $x'_{3,1} \leq x_{3,1}^0 \leq x'_{3,1} + \Delta_3, x'_{4,1} \leq x_{4,1}^0 \leq x'_{4,1} + \Delta_4$. 而由式(17)知, 此时的 $x'_{2,1} \geq x_{2,1}^0 \geq 0$, 因此 $(x'_{2,1}, x'_{3,1}, x'_{4,1})$ 为问题的可行解, 证毕.

设 U_{\max}^{USMS} 为由算法 USMS 求出的问题的近似最优解, U'_{\max} 为可行解 $(x'_{2,1}, x'_{3,1}, x'_{4,1})$ 对应的效用值, 则:

误差

$$\begin{aligned} \delta &= |U_{\max}^0 - U_{\max}^{\text{USMS}}| \leq |U_{\max}^0 - U'_{\max}| \\ &= |V_2 n_2 (u_{2,1}^0 - u'_{2,1}) + V_3 n_3 (u_{3,1}^0 - u'_{3,1}) \\ &\quad + V_4 n_4 (u_{4,1}^0 - u'_{4,1})| \\ &\leq V_2 n_2 (u'_{2,1} - u_{2,1}^0) + V_3 n_3 (u_{3,1}^0 - u'_{3,1}) \\ &\quad + V_4 n_4 (u_{4,1}^0 - u'_{4,1}) \\ &\leq V_2 n_2 \left(\frac{1}{1 + \left(\frac{1}{\epsilon} - 1 \right)^{1 - \frac{2x'_{2,1}}{b_{\max 2,1}}}} - \frac{1}{1 + \left(\frac{1}{\epsilon} - 1 \right)^{1 - \frac{2x_{2,1}^0}{b_{\max 2,1}}}} \right) \\ &\quad + V_3 n_3 \frac{\log((x_{3,1}^0 + 1)/(x_{3,1}^0 + 1 - \Delta_3))}{\log(b_{\max 3,1} + 1)} \\ &\quad + V_4 n_4 \frac{\log((x_{4,1}^0)/(x_{4,1}^0 - \Delta_4))}{\log(b_{\max 4,1}/b_{\min 4,1})} \end{aligned} \quad (20)$$

显然, 当 Δ_3 和 Δ_4 远小于 $x_{3,1}^0, x_{4,1}^0$ 时, 式(20)的后两项远小于 $V_3 n_3 u_{3,1}^0$ 和 $V_4 n_4 u_{4,1}^0$. 对于第一项, 由于

$$\begin{aligned} m \cdot x_{2,1}^0 + \sum_{j=1}^{n_3} ((x_{3,1}^0 + 1)^{p_j}) + \sum_{j=1}^{n_4} (b_{\min 4,j} (x_{4,1}^0/b_{\min 4,1})^{q_j}) &= C_1 \\ m \cdot x_{2,1}' + \sum_{j=1}^{n_3} ((x'_{3,1} + 1)^{p_j}) + \sum_{j=1}^{n_4} (b_{\min 4,j} (x'_{4,1}/b_{\min 4,1})^{q_j}) &= C_1 \\ x'_{2,1} &= x_{2,1}^0 + \Delta_2 \end{aligned} \quad (21)$$

$$\Delta_2 = \frac{1}{m} \left(\sum_{j=1}^{n_3} ((x_{3,1}^0 + 1)^{p_j} - (x'_{3,1} + 1)^{p_j}) + \sum_{j=1}^{n_4} (b_{\min 4,j} ((x_{4,1}^0 / b_{\min 4,1})^{q_j} - (x'_{4,1} / b_{\min 4,1})^{q_j}) \right) \geq 0 \quad (22)$$

考虑到通常同一效用类不同服务的 $b_{\max i,j}$ 的值差异不会很大, p_j, q_j 近似为 1. 因此式(22)近似表明 Δ_2 为 Δ_3 和 Δ_4 的线性组合.

由式(20)(21), 当 Δ_2 远小于 $x_{2,1}^0$ 时可以保证式(20)第一项的值也远小于 $V_2 n_2 u_{2,1}^0$. 因此, 当 Δ_3 和 Δ_4 远小于 $x_{2,1}^0, x_{3,1}^0, x_{4,1}^0$ 时 δ 远小于 U_{\max}^0 .

由以上分析可知, 通过 USMS 算法得到模型的近似最优解.

5.3 改进算法及收敛性证明

考虑到效用函数 $u_{i,j}$ 在 $b_{\min i,j}$ 和 $b_{\max i,j}$ 间连续, 因此模型的最优解很可能存在于 $x_{3,1}$ 的 Δ_3 和 $x_{4,1}$ 的 Δ_4 的领域中. 这样可在此领域中缩小步长进一步进行查找.

USIM 算法是一个迭代过程. 每趟迭代在上次迭代得到的 $x_{3,1}, x_{4,1}$ 的 Δ_3, Δ_4 临域内以 $\Delta_3/\alpha, \Delta_4/\alpha$ 的步长进行搜索. 搜索的过程与 USMS 算法相同. 可以证明该算法的迭代过程是收敛的.

定理 2 USIM 算法的迭代过程是收敛的.

证明 设第 k 趟得到的最优解为 $U_{\max k}$, 此时 $x_{3,1}, x_{4,1}, \Delta_3, \Delta_4$ 的值为 $x_{3,1}^k, x_{4,1}^k, \Delta_3^k, \Delta_4^k$. 考虑到 α 为整数, 且在第 $k+1$ 趟迭代中 x_{i3}, x_{i4} 的初值分别为 $x_{3,1}^k - \Delta_3^k, x_{4,1}^k - \Delta_4^k$, 故 $(x_{3,1}^k, x_{4,1}^k)$ 必出现在 USMS 算法 5~7 步的计算中, 于是 $U_{\max k}$ 成为第 $k+1$ 趟迭代中的候选最优解, 于是有 $U_{\max k+1} \geq U_{\max k}$. 上述分析表明在迭代过程中得到的最优解序列是单调增的, 且有界(因为效用函数是有界的), 因此序列有极限, 算法收敛, 证毕.

算法 2 迭代修正算法 USIM

输入: $U_{\max}, \Delta_3, \Delta_4, x_{2,1}, x_{3,1}, x_{4,1}$, 步长缩减因子 $\alpha (\alpha > 1, \alpha$ 为整数), 变化阈值 ϵ

输出: 优化后的 U_{\max} 和此时 $x_{2,1}, x_{3,1}, x_{4,1}$.

1. $\delta_3 = \Delta_3/\alpha, \delta_4 = \Delta_4/\alpha, U_{\text{old}} = U_{\max}$.
2. $\max x_{3,1} = x_{3,1} + \delta_3, \max x_{4,1} = x_{4,1} + \delta_4$.
3. 令 x_{i3} 的初值为 $x_{3,1} - \Delta_3$.
4. 令 x_{i4} 的初值为 $x_{4,1} - \Delta_4$.
5. 利用式(19)计算 $x_{2,1} \rightarrow x_{i2}$.
6. 如果 $x_{i2} \geq 0$ 利用式(2)~(4)和(12)计算当前的 U , 否则转 9.
7. 如果 $U > U_{\max}, U_{\max} = U$ 存储 x_{i2}, x_{i3}, x_{i4} 到 $x_{2,1}, x_{3,1}$ 和 $x_{4,1}$.
8. $x_{i4} = x_{i4} + \delta_4$, 如果 $x_{i4} \leq \max x_{4,1}$ 转 5.
9. 如果 4~8 步的循环中 $x_{2,1}$ 均小于 0 转 11.
10. $x_{3,1} = x_{3,1} + \delta_3$, 如果 $x_{3,1} \leq \max x_{3,1}$ 转 4.
11. 如果 $(U_{\max} - U_{\text{old}})/U_{\text{old}} > \epsilon, \Delta_3 = \delta_3, \Delta_4 = \delta_4$ 转 1.
12. 返回当前的 $U_{\max}, x_{2,1}, x_{3,1}, x_{4,1}$.

6 实验

为了验证文中模型和模型求解算法的有效性, 设计了仿真实验. 实验中的参数值设置如下: 链路总容量 $C = 10^{10}$ bps, 网络服务总数 $N = 2810$, 参照文献[8], 设 VoIP、IPTV、TCP 和 HTTP 效用类的服务比例分别为 1:3:1:5, $b_{\max 2} = b_{\max 3} = b_{\max 4} = 10^7$ bps, 各个网络服务的带宽需求 $a_{i,j}$, 带宽的最大需求 $b_{\max i,j}$ 按照指定的拥塞度 (CD, Congestion Degree) 在合理的范围内随机产生.

实验的硬软件环境为: Intel® Core™2 Duo T8100 CPU、2G 内存、Windows® Vista 操作系统, USMS 算法和 USIM 算法由 C++ 编程实现, 使用 Microsoft C++ 编译器编译通过.

6.1 U-S 模型与 UBSFTP 模型效用值的比较

在不同拥塞度下, 与 UBSFTP 模型的效用值进行对比, 结果表明 U-S 模型的具有较高的有效性. 实验中各模型的最优值通过 Lingo9.0 中非线性优化算法求得.

因为 VoIP 效用类的带宽需求非常小, 这个效用类的需求会首先被满足(图 3(a)). 在图 3(b), 当 $CD = 2$ 时, UBSFTP 模型中的 IPTV 效用类的效用值降低到 0; 而在 U-S 模型中, 当 $CD = 5$ 时, 其效用值才降低为 0, 表明在 U-S 模型中拥塞的产生要比在 UBSFTP 模型中发生得晚. 因为所有效用类的效用函数曲线都是非递减的, 且 HTTP 效用类在分配与其它效用类相同的带宽时获得的效用值最大, 因此 UBSFTP 模型中的 IPTV 效用类的带宽被分配给了 HTTP 效用类. 当 $CD = 3, 4$ 时 U-S 模型中的 HTTP 效用类所获的效用值比 UBSFTP(图 3(d))模型中对应的效用值小.

从图 4 中可以看出, 在任意拥塞度下, U-S 模型获得的总效用都要比 UBSFTP 模型的大, 平均增加 22.7%, 最大时达 50.58%.

6.2 USMS 算法和 USIM 算法的有效性效率

利用 USMS 算法、USIM 算法和 Lingo9.0 中的相应算法, 分别在拥塞度 1 到 10 的条件下对问题进行求解, 对比所得的效用值和求解时间, 实验中 Δ_3, Δ_4 的取值均为 1000, 图 5 和图 6 为实验结果.

图 5 中的结果显示, 随着拥塞度的增加, 服务总效用随之下降. 通过将 USMS 算法在应用 USIM 算法前后求解得到的效用值与 Lingo 9.0 中算法求得的结果进行对比可以发现, 二者十分接近. 应用 USIM 算法后的效用值平均只比 Lingo 9.0 中算法求得的效用值低 0.4%. 由于通过迭代过程进行了更进一步地搜索, 应用 USIM 算法后所得的效用值比应用前有所提高, 大约平均高 1.1%.

图 6 表明, USMS 算法所消耗的时间远远低于 Lingo9.0 的求解时间, 大约只有后者的 1/10000. 相对于

Lingo 9.0 中的算法, USIM 算法也具有较高的效率. USMS 算法在应用了 USIM 算法后平均耗时约为 Lingo 9.0 求解时间的 1/5000, 表明 USIM 算法收敛速度较快. 从图 6 中还可以发现, 对于不同的拥塞度, Lingo 9.0 中算法耗时变化较大, 而 USMS 算法应用 USIM 算法前后曲线都很平稳, 波动较小, 表明本文提出的算法具有良好的稳定性.

6.3 实验结论

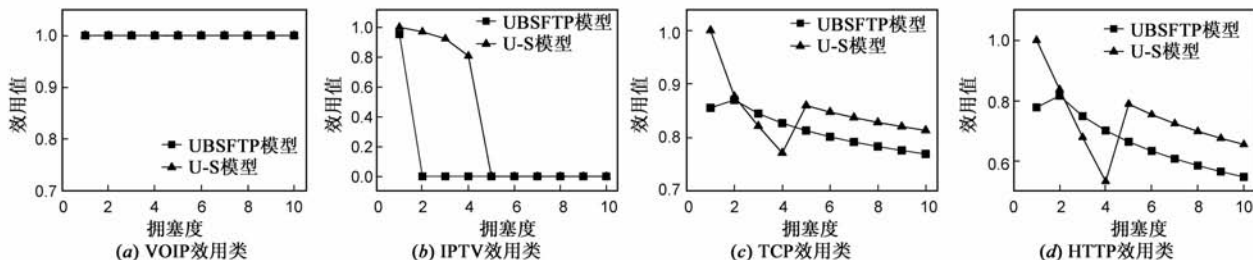


图3 不同的拥塞度下效用值的比较

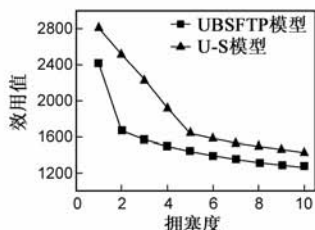


图4 不同拥塞度下模型的总效用值的比较

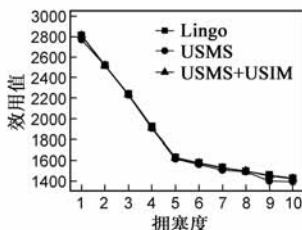


图5 各算法最优效用值的对比结果

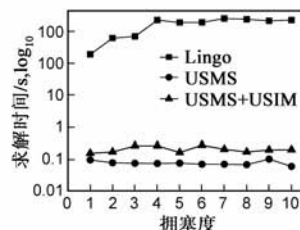


图6 各算法求解时间(log10)的对比结果

7 结论

为了解决传统的基于效用的调度模型(如:单层效用模型 UBSFTP)无法有效地区分同一效用类下服务的不同特征、无法保证基于效用公平的同时实现模型总效用最大化的问题, 本文提出了 U-S 模型, 并给出了求解算法 USMS 和 USIM. 通过仿真实验, 在模型上与 UBSFTP 作了比较, 在运行速度和求解精度上与 Lingo 9.0 中的非线性问题求解算法进行了对比, 结果表明了模型和算法的有效性和高效性. 未来将对效用函数进行进一步的研究和归类, 提出更为通用的模型及其求解算法.

参考文献

- [1] Shenker S. Fundamental design issues for the future internet [J]. IEEE Journal on Selected Areas in Communications, 1995, 13(9): 1176 - 1188.
- [2] Buyya R, Ranjan R, Calheiros R N. Intercloud: utility-oriented federation of cloud computing environments for scaling of application services [A]. Proceedings of International Conference on Algorithms and Architectures for Parallel Processing [C]. Busan, KR: Springer, 2010. 21 - 23.

根据实验结果可以得到以下结论:

- (1) 相对于单层的 UBSFTP 模型, 本文提出的 U-S 模型可取得更高的总效用值;
- (2) USMS 算法可以高效地求得 U-S 模型的近似最优解;
- (3) USIM 算法通过不断减小搜索步长可以进一步提高 USMS 算法解的精度, 而且算法的收敛速度较快.

- [3] 李世勇, 秦雅娟, 张宏科. 基于网络效用最大化的一体化网络服务层映射模型 [J]. 电子学报, 2010, 38(4): 282 - 289.
- Li Shiyong, Qin Yajuan, Zhang Hongke. Mapping model for the service layer of universal network based on network utility maximization [J]. Acta Eletronica Sinica, 2010, 38(4): 282 - 289. (in Chinese)
- [4] 陈明欣, 刘干, 朱光喜. WLAN 中基于效用的呼叫接纳控制策略 [J]. 电子学报, 2008, 36(7): 1429 - 1434.
- Chen Mingxin, Liu Gan, Zhu Guangxi. Utility based call admission control in WLAN [J]. Acta Eletronica Sinica, 2008, 36(7): 1429 - 1434. (in Chinese)
- [5] Kastrinogiannis T, Papavassiliou S. Utility based short-term throughput driven scheduling approach for efficient resource allocation in CDMA wireless networks [J]. Wireless Personal Communications, 2010, 52(3): 517 - 535.
- [6] Cao Z, Zegura E. Utility max-min: an application-oriented bandwidth allocation scheme [A]. Proceedings of IEEE INFOCOM [C]. New York, USA: IEEE Press, 1999. 793 - 801.
- [7] Harks T. Utility proportional fair bandwidth allocation: an optimization [A]. Proceedings of International Workshop on QoS in Multiservice IP Networks [C]. Catania, IT: Springer, 2005. 61 - 74.

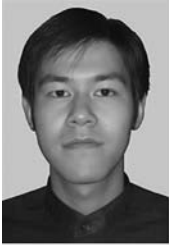
[8] Liu ChangBin, Shi Lei, Liu Bin. Utility-based bandwidth allocation for triple-play services [A]. Proceedings of the European Conference on Universal Multiservice Networks [C]. Toulouse, FR: IEEE Press, 2007. 327 – 336.

[9] Satya R, Liu ChangBin, Liu Bin. Max-min utility fairness in link aggregated system [A]. Proceedings of the International Workshop on High Performance Switching and Routing [C]. New York, USA: IEEE Press, 2007. 954 – 962.

作者简介



宋亚楠 女, 1977 年生于河南虞城, 清华大学计算机科学与技术系博士生. 研究方向为网络资源调度和服务质量.
E-mail: song-yn07@mails.tsinghua.edu.cn



胡成臣 男, 1981 年 8 月出生, 出生于浙江杭州, 西安交通大学电信学院计算机系副教授. 研究方向为网络测量.



仲 茜 男, 1975 年生于山东济南, 2010 在清华大学计算机科学与技术系取得工学博士学位, 研究方向为语义数据集成、数据搜索和语义 Web.



刘 斌 男, 清华大学计算机科学与技术系教授, 博士生导师. 研究方向为网络交换与调度、下一代互联网.