

P2P 分层流媒体数据分配的粒子群遗传算法

黄继海, 杨志宏, 赵建勋

(中州大学信息工程学院, 郑州 450044)

摘 要: 现有 P2P 分层流媒体中的数据分配算法是基于贪婪思想的确定性启发式算法, 不能得到全局最优解。为此, 提出一种基于备选数据块编码方式的粒子群遗传算法。定义备选数据块, 建立问题的无约束整数规划模型。仿真实验表明, 该算法在优化效果上能比现有算法提高 5%~25%。

关键词: 对等网络; 分层流媒体; 数据分配; NP 完全问题; 粒子群遗传算法

Particle Swarm Genetic Algorithm of Data Allocation in P2P Layered Streaming Media

HUANG Ji-hai, YANG Zhi-hong, ZHAO Jian-xun

(Information Engineering College, Zhengzhou University, Zhengzhou 450044, China)

【Abstract】 Data allocation in layered P2P streaming media is proved to be a NP-complete problem. The existing algorithm is a heuristic algorithm based on the greedy idea, which can not get the global solution. In order to develop a new algorithm for this problem, it defines the concept of data blocks for choosing, sets up a mathematic model of integer programming without restriction, and proposes a novel Particle Swarm Genetic Algorithm(PSGA) based on the encoding manner of Data Blocks for Choosing. Simulation demonstrates that the proposed PSGA's performance improves 5%~25% than that of the existing algorithm.

【Key words】 Peer-to-Peer(P2P) network; layered streaming media; data allocation; NP-complete problem; Particle Swarm Genetic Algorithm (PSGA)

DOI: 10.3969/j.issn.1000-3428.2011.17.063

1 概述

基于单播 C/S 结构的媒体服务器容易成为系统瓶颈, 其扩展性不高, 不适合流媒体大规模数据的分发。P2P 流媒体技术成为研究热点^[1]。P2P 分层流媒体中的数据分配问题, 按照粒度可分为 2 类, 包括粗粒度调度和细粒度调度。其中, 细粒度调度是基于数据包的调度算法, 目前已有比较丰富的研究成果, 粗粒度调度是基于数据层的调度算法, 目前只有极少量的研究报告。文献[2]研究了在 P2P 分层流媒体数据分配中公平限制的条件下获得最好质量流媒体的路由问题; 文献[3]设计了一个基于“贡献上行带宽越多得到数据层数多”激励机制的协议。文献[4]分别针对分层编码速率同构与异构这 2 种情况提出了 2 类数据分配算法: (1)在不限制数据供应节点数前提下的数据层分配算法; (2)在限制数据供应节点数前提下的数据层分配算法。文献[5]在文献[4]的基础上仅假设异构的分层编码速率, 实际上该假设更具普遍性^[6]。文献[5]建立了数据分配问题的非线性整数规划模型, 证明了这个问题为 NP-complete 问题, 并且设计了 2 个启发式算法 HA-WOR 和 HA-WR 用于解决该问题。但是, 文献[5]所建的数学模型含有复杂的非线性约束条件, 其设计的 HA-WOR 和 HA-WR 算法是基于“贪婪法思想”的确定性启发式算法, 不能保证得到问题的最优解。

本文的工作属于对文献[5]工作的改进和深入。为了设计性能更好的数据分配算法, 本文定义了“备选数据块”的概念, 并建立了数据分配问题的非线性整数规划模型, 此模型不含约束, 在形式上比文献[5]的模型更简单。然后, 基于备选数据块的编码方式, 设计了一个改进的粒子群遗传算法,

由于采用了更简单的数学模型和基于备选数据块的编码方式, 交叉和变异产生的每一个新个体都是合法的, 算法的搜索效率得到了保证。

2 数学模型

标记 Root 节点为 p_0 , 并假设它有充足的带宽资源。节点 q 所请求的流媒体为 s , s 的最大编码层数为 L_s , 第 i 个数据层的编码速率为 $r_i(i=1,2,\dots,L_s)$ 且速率不变, 第 i 个数据层的解码依赖于所有第 1 层~第 $(i-1)$ 层数据的完全解码。 q 的下行带宽为 B_q , q 能够获取的最大数据层数 m 受 B_q 的限制, m 的值可由下式确定:

$$m = \max \left\{ \chi \mid \sum_{i=1}^{\chi} r_i \leq B_q, 1 \leq \chi \leq L_s \right\} \quad (1)$$

定义 1(可用节点) 如果系统中某个 Peer 节点 p 包含节点 q 所请求的数据层, 且 p 到 q 的可用带宽不小于 $\min(r_i)(i=1,2,\dots,m)$, 则称 p 为 q 的可用节点。

设 q 的可用节点集合为 S_q , S_q 的成员个数为 n , 其成员序列为 p_1, p_2, \dots, p_n 。 $p_j(j=1,2,\dots,n)$ 上所缓存的流媒体数据层数为 $a_j(1 \leq a_j \leq L_s)$, p_j 到 q 的上行带宽为 b_j 。显然, p_j 为节点 q 所提供的最大数据层数不能超过 a_j , 速率之和不能超过 b_j 。

基于数据层的调度问题, 按照 Root 节点 p_0 是否参与亦可分为 2 种情况:

基金项目: 河南省科技攻关计划基金资助项目(102102210247)

作者简介: 黄继海(1977—), 男, 讲师、CCF 高级会员, 主研方向: 人工智能, 分布式计算系统; 杨志宏, 副教授; 赵建勋, 讲师

收稿日期: 2011-03-03 **E-mail:** huangjhai@sina.com

(1)Root 节点 p_0 不参与的情形, 请求节点 q 只能从可用节点集 S_q 处下载数据, 其优化目标是使 q 获得从第 1 层开始的尽可能多的连续数据层, 以尽可能提高 q 的视频播放质量。

(2)Root 节点 p_0 参与的情形, 通常服务节点 p_0 有足够的带宽资源, 因此 q 的播放质量不再是主要问题, 问题的优化目标是如何让普通节点 p_1, p_2, \dots, p_n 承担尽可能多的数据层分发任务, 从而减轻 Root 节点的负担。

以上 2 类问题都已被证明是 NP 完全问题^[5], 这意味着不存在解决上述问题的确定性多项式时间算法。

定义 2(可用数据块) 对于连续数据层 $B_j = L_u \cup L_{u+1} \cup \dots \cup L_v$, $1 \leq u \leq v \leq a_j$, 若其对应的编码速率之和不大于其上行带宽, 则称 B_j 为 p_j 的可用数据块, 令 W 表示 p_j 的可用数据块的总个数, 称 $\Phi_j = \{B_j^1, B_j^2, \dots, B_j^W\}$ 为 b_j 的可用数据块的集合。

定理 1 b_j 的可用数据块的个数小于等于 $W = a_j(a_j+1)/2$ 。

证明: 按照含有的数据层的个数, 对 b_j 的可用数据块进行分类: 只含有 1 个数据层的数据块有 a_j 个, 含有 2 个连续数据层的数据块有 (a_j-1) 个, \dots , 含有 a_j 个连续数据层的数据块只有 1 个, 总共有 $1+2+\dots+a_j = a_j(a_j+1)/2$ 个数据块。以上是不考虑上行带宽的情况, 如果再考虑带宽约束, 可用数据块的个数将小于等于 $a_j(a_j+1)/2$ 个, 得证。

可用数据块之间存在包含与被包含的关系, 不妨将所有存在包含与被包含关系的可用数据块进行归并处理, 删掉能被其他数据块包含的小数据块, 最后只剩下若干大的数据块, 这些剩下的可用数据块被称为备选数据块。

定义 3(备选数据块) 对于任意的 $B_j^w \in \Phi_j$, 如果不存在 $B_j^w(i=1, 2, \dots, W)$ 且 $i \neq w$, 使得 $B_j^w \subset B_j^i$, 则称 B_j^w 为 p_j 的备选数据块, 称 $\Gamma_j = \{B_j^1, B_j^2, \dots, B_j^T\}$ 为 b_j 的备选数据块的集合。

定理 2 b_j 的备选数据块的个数最多不会超过 a_j 。

证明: (穷举法) 含有 a_j 个连续数据层的可用数据块只有 1 个, 其他的可用数据块都被它包含, 如果它被判定为备选数据块, 那么备选数据块只有这 1 个; 除此之外, 再考虑含有 (a_j-1) 个连续数据层的可用数据块, 最多只有 2 个这样的可用数据块, 且它们互不包含, 如果它们被判定为备选数据块, 那么备选数据块最多只有这 2 个; \dots ; 依此类推, 排除以上各类情况后, 最后考虑含有 1 个连续数据层的可用数据块, 共有 a_j 个这样的可用数据块, 且它们互不包含, 如果它们被判定为备选数据块, 那么备选数据块只有这 a_j 个, 得证。

令 $X = [x_1, x_2, \dots, x_n]$, 其中, $1 \leq x \leq T_j$, T 表示第 j 个节点最大备选数据块数; 其意义为 b_j 的第 x_j 个备选数据块被选中。用 D 表示来自不同可用节点的备选数据块的并集, 即:

$$D = \bigcup_{j=1}^n B_j^{x_j} \quad (2)$$

定义 4(映射 f_A) 运算 $f_A(D)$ 表示取 D 的从第 1 层开始的连续数据层的总层数。

定义 5(映射 f_B) 运算 $f_B(D)$ 表示 D 的数据层对应的编码速率之和。

在定义 4 和定义 5 的基础上可以建立 P2P 分层流媒体数据分配的数学模型为:

$$\min f_A(D) \text{ or } f_B(D) \quad \text{s.t. } D = \bigcup_{j=1}^n B_j^{x_j}, 1 \leq x_j \leq T_j \quad (3)$$

如果式(3)中的目标函数为 f_A , 则表示 Root 节点不参与的情况; 如果目标函数为 f_B , 则表示 Root 节点参与的情况。此模型是一个无约束非线性整数规划模型, 可以使用粒子群遗传算法求解, 下面介绍本文设计的粒子群遗传算法。

3 粒子群遗传算法

3.1 种群的构造

本文设计的粒子群遗传算法采用备选数据块的编码方式, 个体表示为一个向量 $X = [x_1, x_2, \dots, x_i, \dots, x_n]$, 其中, x_i 是介于 1 和 T_j 之间的一个整数; T_j 是节点 j 的备选数据块的总个数。在粒子群遗传算法运行之前, 要构造基于备选数据块的种群, 可使用如下算法构造备选数据块:

Step1 断开: 对于第 j 个 peer 节点缓存的 a_j 层流媒体数据, 剔除单层编码速率超过其上行带宽的数据层, 从而形成若干个断开连续数据层的集合。

Step2 独立搜索: 对于某个断开的连续数据层的集合 D , 假设其高度为 H , 执行下面的搜索可以找出全部备选数据块。

```
for p1=1:H
for p2=p1:H;
if r(p1)+r(p1+1)+...+r(p2)≤b_j && r(p1)+r(p1+1)+...+r(p2+1)>b_j
将 D(p1)到 D(p2)的连续数据层加入备选数据块集合
end
if p2==H;
return;
end
end
end
```

该算法时间复杂度在最不利的情况下为 $O(a_j^2)$, 在最有利的情況下为 $O(a_j)$ 。

该算法的原理如图 1 所示。

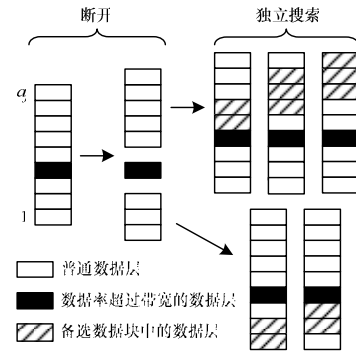


图 1 产生备选数据块的示意图

3.2 编解码系统

根据 3.1 节中算法的输出矩阵 D 可以得知节点的备选数据块集合, 用决策变量 $X = [x_1, x_2, \dots, x_i, \dots, x_n]$ 表示染色体, 下面用图示的方法对编解码系统作一个说明, 图 2 给出了编码系统的示意图, 图 3 给出了解码系统的示意图。

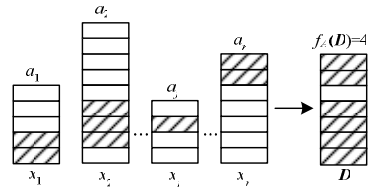


图 2 编码示意图

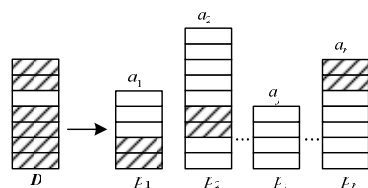


图 3 解码示意图

编码的过程就是从各个 peer 节点的备选数据里各挑选出一个备选数据块, 然后把这些备选数据块拼在一起, 得到一个大的数据层的集合, 解码的过程与之相反。编码的过程是多对一的映射, 解码的过程则是一对多的映射。

3.3 交叉方式

对于备选数据块的编码方式, 本文采用双亲双单点交叉方式产生新个体。下面举例说明, 设父代染色体 $A=32542633$ 、 $B=24365721$, 随机选择第 4 位和第 5 位基因之间的点为交叉点, 交叉之后得到子代染色体为 $A=32545721$, $B=24362633$, 如图 4 所示。

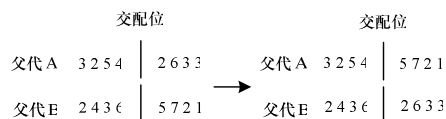


图 4 双亲双单点交叉

3.4 变异算子

变异算子具体描述如下:

Step1 产生随机数 $rand$, 如果变异概率 $P_m > rand$, 则执行下面的操作, 否则什么也不做。

Step2 根据式(2)和式(3)计算评价函数 $f(D)$ 。

Step3 对 D 实施变异操作, 即随机地更换一个基因, 得到一个新个体。

Step4 根据式(2)和式(3)计算新的评价函数 $f(x')$, 并计算评价函数的差值 $\Delta f = f(x') - f(x)$ 。

Step5 若 $f(x') > f(x)$, 说明新个体优于旧个体, 新个体替换旧个体; 若 $f(x') \leq f(x)$, 说明新个体劣于旧个体, 但仍然以概率 $\exp(\Delta f/T)$ 替换旧个体。

4 仿真实验与分析

针对以上约束条件以及目标函数设计编码序列模型, 在 Matlab7.0 环境下, 使用 PSGA 算法进行仿真, 运行环境是 Pentium Dual E2104, 1.6 GHz, 512 MB, Microsoft XP。参数设置如下: (1)基本遗传算法: 最大进化代数 300, 种群规模为 20, 交叉率为 0.85, 变异率为 0.005; (2)MPS 算法: 最大进化代数 200, 学习因子分别为 $c1=2$, $c2=1.8$, 惯性权重因子 w 从 2 降低到 0.8, 扰动因子 $u=10$, 最大速度为 4。

4.1 实验数据

在仿真实验中, 本文对源节点不参与时请求节点可以获得第 1 层开始的连续数据层的总层数(即 $f_A(D)$, 以下简称目标 A), 和源节点参与时请求节点可以获得的数据层对应的编码速率之和(即 $f_B(D)$, 以下简称目标 B)进行了评估。在分层编码模式(其编码速率从低层到高层依次分别为 {20,20,96,64,32,128,160,192,128,64} Kb/s)的情形下, 设请求节点的下行带宽为 600 Kb/s; 数据提供节点上行带宽为 20 Kb/s~300 Kb/s 中的随机值, 缓存的节点层数为 1~10 之间的随机值。

4.2 实验结果

针对数据提供节点数为 2~50 时求解目标 A 和目标 B 2 个实验均重复运行 600 次。对 600 次实验得到的 $f_A(D)$ 和 $f_B(D)$ 的平均值与文献[5]提出的算法进行比较。2 个实验仿真结果分别如图 5、图 6 所示。由图 5 可以看出, 求解目标 A 时, 在数据提供节点数范围为 2~30 时 PSGA 比 SCA-WOR 性能提高 5%左右, 在数据提供节点数范围为 31~50 时两算法性能一致。由图 6 可以看出, 求解目标 B 时, 在数据提供节点数范围为 2~30 时, PSGA 比 SCA-WOR 性能提高 25%左右,

在数据提供节点数范围为 31~50 时, PSGA 比 HA-WR 性能提高的比例有缩小的趋势。

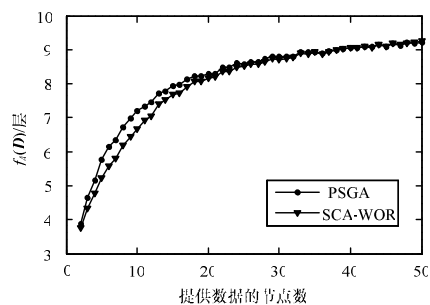


图 5 请求节点获得的最大连续数据层数

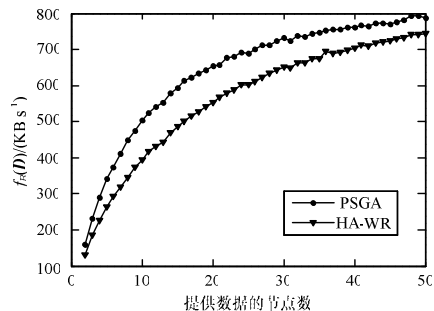


图 6 请求节点获得最大连续数据层的编码速率和

这 2 个实验表明, 在数据提供节点数较多的情况下, 请求节点有更多的选择余地, 所以 SCA-WOR 和 HA-WR 的性能有接近 PSGA 的趋势。但是, 在实际系统中, 如果数据提供节点数越多, 那么系统就要付出维护请求节点与数据提供节点之间关系的代价就越多。因此, PSGA 在实际系统中更具实用价值。

5 结束语

针对 P2P 网络中节点资源能力的异构性的问题, 本文研究了 P2P 分层流媒体中基于数据层的分配问题, 提出一种解决 P2P 分层流媒体中数据层分配问题的算法, 在优化效果上比现有算法有所提高, 而且该算法的收敛性较好, 从而为这类问题提供了一种新的更好的解决方案。

参考文献

- [1] 胡平, 聂朋朋. 典型 P2P 流媒体模型及其关键技术[J]. 计算机工程, 2009, 35(3): 60-62.
- [2] Dai Liang, Cui Yi. Maximizing Throughput in Layered Peer-to-Peer Streaming[C]//Proc. of ICC'07. Glasgow, UK: [s. n.], 2007: 1734-1739.
- [3] Liu Zhengye, Shen Yanming. Using Layered Video to Provide Incentives in P2P Live Streaming[C]//Proc. of ACM P2P-TV'07. Kyoto, Japan: [s. n.], 2007: 311-316.
- [4] Cui Yi, Nahrstedt K. Layered Peer-to-Peer Streaming[C]//Proc. of ACM NOSSDAV'03. New York, USA: ACM Press, 2003.
- [5] 刘亚杰, 张鹤颖. P2P 分层流媒体中数据分配算法[J]. 软件学报, 2006, 17(2): 325-332.
- [6] Kim T, Ammar M. A Comparison of Layering and Stream Replication Video Multicast Scheme[C]//Proc. of ACM NOSSDAV'01. New York, USA: ACM Press, 2001: 63-72.

编辑 任吉慧

