

基于蚁群算法的遥感影像传输资源调度方法

刘万军¹, 王晓宇^{1*}, 曲海成^{1,2}, 孟煜¹, 姜庆玲³

(1. 辽宁工程技术大学 软件学院, 辽宁 葫芦岛 125105; 2. 哈尔滨工业大学 电子与信息工程学院, 哈尔滨 150006;

3. 铁岭师范高等专科学校 工学院, 辽宁 铁岭 112008)

(* 通信作者电子邮箱 410867038@qq.com)

摘要:针对遥感影像数据量大,多用户并发请求造成服务器负载加重,使遥感影像传输效率逐渐降低的问题,提出一种在多线服务器环境下分块调度遥感影像资源的策略。该策略采用改进的蚁群优化(IACO)算法,通过引入一个线路等待因子 γ 动态选择当前最优的线路进行传输,从而提高传输效率。对IACO、ACO、Max-min、Min-min和Random算法进行了对比实验,IACO算法在客户端的任务完成时间和服务器端的执行时间与其他算法相比均是最少的,且随着任务数目的增加,效果更明显;同时IACO算法的线路资源的利用率也更高。仿真结果表明:多线服务器分块调度策略与改进蚁群算法相结合,使遥感影像传输速度和线路资源利用率均有一定提高。

关键词:遥感;资源调度;蚁群优化算法;多线服务器;资源利用率

中图分类号: TP75 **文献标志码:** A

Transmission resource scheduling method for remote sensing images based on ant colony algorithm

LIU Wanjun¹, WANG Xiaoyu^{1*}, QU Haicheng^{1,2}, MENG Yu¹, JIANG Qingling³

(1. Software Engineering Institute, Liaoning Technical University, Huludao Liaoning 125105, China;

2. School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin Heilongjiang 150006, China;

3. Engineering Institute, Tieling Normal College, Tieling Liaoning 112008, China)

Abstract: A block resource scheduling strategy for remote sensing images in multi-line server environment was proposed with the problems of huge amount of remote sensing data, heavy server load caused by multi-user concurrent requests which decreased the transmission efficiency of remote sensing images. To improve the transmission efficiency, an Improved Ant Colony Optimization (IACO) algorithm was used, which introduced a line waiting factor γ to dynamically select the optimal transmission lines. Intercomparison experiments among IACO, Ant Colony Optimization (ACO), Max-min, Min-min, and Random algorithm were conducted and IACO algorithm finished the tasks in the client and executed in the server with the shortest time, and the larger the amount of tasks, the more obvious the effect. Besides, the line resource utilization of IACO was the highest. The simulation results show that: combining multi-line server block scheduling strategy with IACO algorithm can raise the speed of remote sensing image transmission and the utilization of line resource to some degree.

Key words: remote sensing; resource scheduling; Ant Colony Optimization (ACO) algorithm; multi-line server; resource utilization

0 引言

随着卫星遥感技术的不断发展,高分辨率遥感影像数据量呈几何级数增长,各研究机构都积累了大量的遥感数据。遥感应用对数据处理的大量需求,促进了遥感影像高性能处理方法和技术的发展。目前服务器多线路接入技术被广泛应用到遥感影像传输系统中^[1]。在多线服务器环境中,把不同的多家互联网服务提供商(Internet Service Provider,ISP)服务接入到一台服务器,使所提供的网络服务访问用户能尽可能以同一个ISP或互访速度较快的ISP连接进行访问,从而解决或者减轻跨ISP用户访问网站的网络瓶颈问题^[2]。然而,

在传统的多线服务器中,当客户端访问服务器时,只能选择其中的一条线路进行传输,传输效率较低。

在多线服务器环境中,使用负载均衡算法^[3-4]可以提高每个资源的利用率,减少总任务的执行时间,尽可能地让线路资源上的任务达到平衡。资源调度是一个NP完全问题,解决此类问题已经有许多算法。常用的算法包括:果蝇优化算法(Fruit fly Optimization Algorithm,FOA)、粒子群优化(Particle Swarm Optimization,PSO)算法、蚁群优化(Ant Colony Optimization,ACO)算法和遗传算法(Genetic Algorithm,GA)^[5-7]。多线服务器中线路资源的占用情况动态变化,这种不确定性给资源调度带来了很大困难,而蚁群算法是一种

收稿日期:2014-06-01;修回日期:2014-07-09。

基金项目:国家863计划项目(2012AA12A405);国家自然科学基金资助项目(61172144)。

作者简介:刘万军(1959-),男,辽宁北镇人,教授,博士生导师,主要研究方向:数字图像处理、运动目标检测与跟踪;王晓宇(1991-),女,山西忻州人,硕士研究生,主要研究方向:遥感影像快速处理;曲海成(1981-),男,山东烟台人,博士研究生,主要研究方向:遥感影像高性能计算;孟煜(1990-),男,河北唐山人,硕士研究生,主要研究方向:图形图像处理、目标识别与跟踪。

启发式的算法,具备解决大规模组合优化问题的能力,其动态更新性能较适合资源调度。

本文针对遥感影像的数据量大、可以分块传输的特点,为了提高传输速度,提高线路资源利用率^[8],提出对一景影像利用多条线路进行传输的资源调度策略,并且结合改进的蚁群优化(Improved Ant Colony Optimization, IACO)算法提高了线路的传输效率。

1 遥感影像传输的资源调度策略

遥感影像传输的资源调度^[9]是客户端与服务端之间相互通信的一个重要环节,其主要作用是通过获得任务和线路资源的实时状态信息进行任务的调度。这样可以更加有效地利用服务端资源,均衡网络负载,提高传输的整体性能,发挥并行环境的高性能优势^[10]。

1.1 遥感影像的传输方式

多个客户端与服务端连接成功后,客户端向服务端发送所需遥感影像的经纬度坐标或区域名称,服务端在数据库中检索客户端所需影像是否存在,如果存在则读取图像。服务端将读取到的图像进行分块,分解成多个子任务。子任务根据线路质量,选择传输速度较快的线路进行传输。如果当前最优线路被占用则选取次优线路,当子任务在次优线路上传输时,如果发现此时最优线路已空闲则立即选取最优线路进行传输,最后经由各个客户端发来请求的线路传回数据。这样在完成一个客户端请求的任务时,通过分块且在多个不同的线路中传输,可以减少传输时间,提高服务器资源的利用率。

选取 n 个客户端和 m 条由不同 ISP 接入的线路,其中客户端 1 向服务器发送请求,服务器计算其子任务个数,通过调度设备将这些子任务配发到各线路中,最后经由 ISP 节点 1 向客户端发送数据。该过程遥感影像传输的框架如图 1 所示。

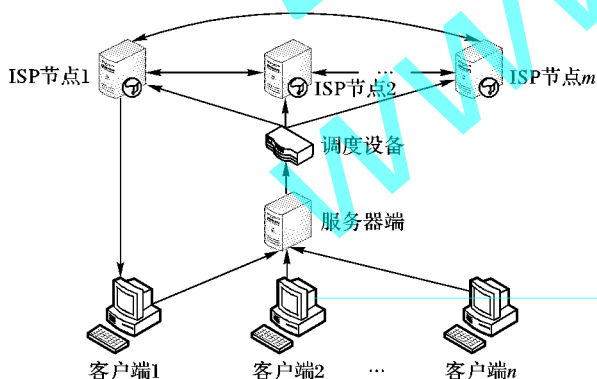


图 1 遥感影像的传输框架

1.2 资源调度策略分析

遥感影像传输中资源调度策略的实质是解决任务与可用线路的匹配问题。每个客户端选取的线路由带宽和排队等待时间所决定,如果在调度过程中客户端选取的线路存在冲突,则由该条线路上带宽最大的客户端优先抢占线路。以下用一个示例来具体说明遥感影像在传输过程中的资源调度策略:

选取 3 个客户端和 3 条线路进行传输,这 3 个客户端单位时间内分别在 3 条线路上传输的数据块数量(此例中用来

代表线路的带宽),以及这 3 个客户端请求的遥感影像数据任务在服务器端被分成的数据块总数即子任务的数量,见表 1。

表 1 客户端数据块单位时间传输个数及总数

线路	客户端 A	客户端 B	客户端 C
1	500	700	800
2	300	400	500
3	100	200	200
总数	3 000	6 000	5 000

资源调度策略的具体方法如图 2 所示。因为客户端 C 在线路 1 上的带宽最大,则选择该线路传输,完成时间为 6.25 s。同理客户端 B 和客户端 A 分别在线路 2 和线路 3 上传输 6.25 s。此时线路 1 空闲由客户端 B 抢占进行传输 5 s,传输完成。这段期间客户端 A 选择线路 2 传输,当线路 1 再次空闲则选择其进行传输。通过这个简单的例子可看出使用多线路进行传输可以使网络负载更加均衡,传输效率更高。

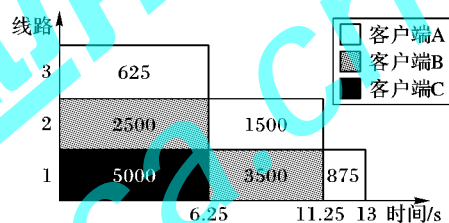


图 2 资源调度策略的具体方法

2 遥感影像传输的资源调度算法

蚁群算法^[11-12]真实地模拟了自然界蚂蚁群体的觅食行为。蚂蚁在寻找食物源时,会在其经过的路上释放一种信息素,并且能够感知其他蚂蚁释放的信息素。通常蚂蚁会以较大的概率选择信息素浓度较高的路径,并释放一定量的信息素,以增强该条路径上的信息素浓度,这样形成了一个正反馈。蚁群算法被广泛地应用到许多 NP 问题中,并且取得了很好的效果。

蚁群算法具有自适应性强、分布式计算、正反馈的特点,可以加快遥感影像传输任务调度的求解。蚁群算法是蚂蚁之间通过利用释放的信息素寻找最优解的问题,有很强的并行性和鲁棒性,对初始化条件的要求不太高,可以适用于复杂的网络传输环境。蚁群算法是一种很好的求解组合优化问题的方法,遥感影像传输的资源调度是用来寻找客户端请求任务与线路资源的最优组合方式,是一种组合优化问题,使用蚁群算法可以解决遥感影像传输的资源调度问题。遥感影像的传输过程也是一种分布式计算,这个过程中资源调度的分配问题可以利用蚁群算法的协作性和并行性更好地解决。

2.1 节点信息素初始化

在 IACO 算法中,引入线路等待因子 γ ,它表示线路资源 j 处于空闲状态所需要的等待时间。通过 γ 可以动态地了解线路的负载情况,在线路的选择和判断中起着关键的作用,使任务的调度效果更好。

在 t 时刻,线路资源的信息素定义为:

$$\tau_j(t) = 1/\gamma_j; 1 \leq j \leq m \quad (1)$$

其中: γ_j 为线路 j 的等待因子,其初始值 $\gamma_j = a$ (a 是常量, $0 <$

$a < 1$); m 为线路资源节点数。

启发函数定义为:

$$\eta_j(t) = B_j; 1 \leq j \leq m \quad (2)$$

其中 B_j 表示线路 j 的带宽。

2.2 对线路资源节点进行选择

在遥感影像传输的资源调度中,每个客户端任务的子任务不能同时出现在多条线路中,这样保证了每个任务的连续性和完整性。把一个数据块当作一只蚂蚁,这些蚂蚁是同一个客户端在同一时刻产生的,称之为同类蚂蚁。同类蚂蚁有相同的序列号,且不能同时出现在多条线路中。每只蚂蚁会按照一定的标准根据线路的质量去选择较优的线路。线路中的转移概率为 p_{ij}^t ,蚂蚁通过计算概率来选取线路,这样任务就会动态地在当前最优的线路上进行传输,与传统的静态地选取一条线路传输在传输效率上形成了鲜明的对比。

p_{ij}^t 表示在 t 时刻第 i 个任务选择线路 j 的概率。 p_{ij}^t 定义为:

$$p_{ij}^t = \begin{cases} \frac{\tau_j^\alpha(t) \eta_j^\beta(t)}{\sum_k \tau_k^\alpha(t) \eta_k^\beta(t)}, & j, k \in L \\ 0, & j, k \notin L \end{cases} \quad (3)$$

其中: L 为所有线路的集合; $\tau_j(t)$ 为 t 时刻线路 j 的信息素浓度,表示蚂蚁转移到线路 j 的期望程度; $\eta_j(t)$ 为 t 时刻线路 j 的启发函数; α 为信息素的重要程度; β 为启发函数的重要程度。

2.3 信息素更新

一个任务传输完成后,各线路的信息素会进行更新,其更新的公式为:

$$\gamma_j = \gamma_j + T_{ij} \quad (4)$$

$$\tau_j(t) = (1 - \rho) / \gamma_j \quad (5)$$

其中: ρ 为信息素的挥发程度, T_{ij} 为第 i 个任务在线路 j 上完成的时间。

2.4 改进蚁群算法流程

步骤1 客户端提交任务,将其插入到任务队列中等待调度,同时服务器端为每个任务计算分块个数即子任务的数量。

步骤2 对资源调度信息初始化,初始化 $\tau, \eta, \rho, \alpha, \beta$ 等参数,利用式(1)~(2)初始化各线路资源节点的初始信息素和启发函数。计算出初始任务在线路中执行所用的时间,也就是下一任务在该线路上的等待时间。

步骤3 将任务队列中的任务,根据式(3)计算出转移概率。每个子任务相当于一只蚂蚁,头蚂蚁在选择线路资源时,通过得出的最大概率来判断此时最优的线路进行调度。在传输的过程中,当有最优线路空闲时,蚂蚁再进行计算和选择新的路线,确定任务-线路匹配。

步骤4 当任务正常分配给线路后,计算出任务 i 在线路 j 上执行的时间 T_{ij} ,利用式(4)~(5)更新信息素 $\tau_j(t)$ 。

步骤5 循环步骤2~4,直到每个客户端的同类蚂蚁都执行完成。

步骤6 判断各客户端的任务是否完成。

步骤7 算法结束,输出结果,整个过程执行完成。

3 实验与分析

由于遥感影像数据在传输的过程中,一幅影像被分割成

了很多的数据块,本文研究的资源调度策略实际解决的是任务对线路资源的分配问题。在进行实验仿真的过程中,可以看作是一个特殊的网格问题,所以使用 Gridsim 来模拟一个遥感影像数据传输的局部区域,以测试算法在这种特殊的网格环境中的运行情况。在 Gridsim 平台上,分别作了 IACO 算法、ACO 算法、Max-min 算法、Min-min 算法、Random 算法的对比实验。蚁群算法中,经过大量实验表明,参数 α 和 β 都取 0.5, ρ 取 0.2 较合适。

实验1 分析客户端任务完成的时间。

实验中计算所有客户端任务完成的总时间,选取 10 条线路资源,客户端的总任务数分别为:500,1000,1500,2000,2500,3000,3500 进行实验。任务完成时间的实验结果如图3所示。对实验结果分析可知,当客户端的任务数目确定时,与其他算法相比,IACO 算法花费的时间最少,随着任务数目的增加,效果更明显。

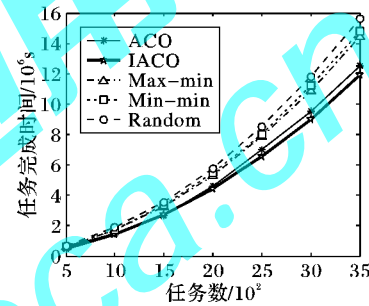


图3 任务完成时间的实验结果

实验2 分析服务器端执行完成的时间。

线路资源数目为 50,任务数目分别为:2000,2500,3000,3500,4000,4500,5000。实验研究服务器端执行的时间,相当于在服务器端最后一个任务传输完成的时间。服务器端执行时间的实验结果如图4所示。实验结果表明任务数越多,改进的蚁群算法在执行完成的时间上明显优于其他算法。

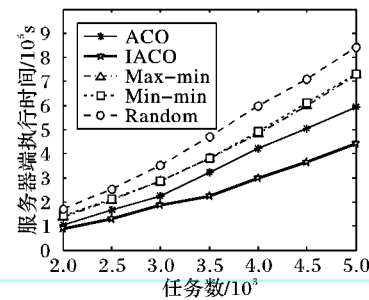


图4 服务器端执行时间的实验结果

实验3 分析线路资源的利用率。

线路资源的利用率定义为:

$$per = \frac{\sum_{i=1}^m T_j}{\max(T_1, T_2, \dots, T_m) \times m} \times 100\% \quad (6)$$

其中: T_j 为第 j 条线路执行任务的时间, $\max(T_1, T_2, \dots, T_m)$ 为所有线路中执行任务最长的时间。选取线路资源数目为 20,客户端总任务数目分别为 1000,2000,3000,4000,5000 进行对比实验。线路资源的利用率的柱状图如图5所示。对实验结果分析可知,改进的蚁群算法优于其他算法,线路资源的利用率比一般的蚁群算法提高了 16% 左右。

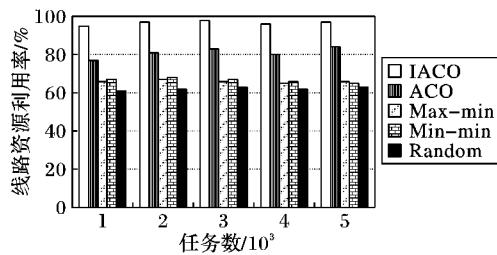


图 5 线路资源利用率柱状图

4 结语

本文提出了一种在多线服务器环境下传输遥感影像的资源调度策略,此策略利用改进的蚁群算法,更好地解决遥感影像数据量大、多线路传输的问题,调度效果明显。改进的蚁群算法使任务传输所用的时间更少,线路资源的利用率提高了 16% 左右,可以更好地利用服务器资源。在以后的研究中,可以将不同的多家 ISP 服务接入到一个服务器集群上面处理遥感数据,在遥感影像传输的资源调度过程中实现负载均衡,大大提高传输效率。

参考文献:

- [1] BLASCHKE T. Object based image analysis for remote sensing[J]. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2010, 65(1): 2-16.
- [2] NYGREN E, SITARAMAN R K, SUN J. The Akamai network: a platform for high-performance Internet applications[J]. *ACM SIGOPS Operating Systems Review*, 2010, 44(3): 2-19.
- [3] LU Y, XIE Q, KLIOT G, et al. Join-Idle-Queue: a novel load balancing algorithm for dynamically scalable Web services[J]. *Performance Evaluation*, 2011, 68(11): 1056-1071.
- [4] GAWANDE D S, DHARMIK R C, PANSE C. A load balancing in grid environment[J]. *International Journal of Engineering Research and Applications*, 2012, 2(2): 445-450.
- [5] FILIPOVIC V. Fine-grained tournament selection operator in genetic algorithms[J]. *Computing and Informatics*, 2012, 22(2): 143-161.
- [6] VALDEZ F, MELIN P, CASTILLO O. An improved evolutionary method with fuzzy logic for combining particle swarm optimization and genetic algorithms[J]. *Applied Soft Computing*, 2011, 11(2): 2625-2632.
- [7] COSSELL S, GUIVANT J. Concurrent dynamic programming for grid-based problems and its application for real-time path planning[J]. *Robotics and Autonomous Systems*, 2014, 62(6): 737-751.
- [8] LEE Y C, ZOMAYA A Y. Energy efficient utilization of resources in cloud computing systems[J]. *The Journal of Supercomputing*, 2012, 60(2): 268-280.
- [9] GORBENKO A, POPOV V. Task-resource scheduling problem[J]. *International Journal of Automation and Computing*, 2012, 9(4): 429-441.
- [10] SHI L, CHEN H, SUN J, et al. vCUDA: GPU-accelerated high-performance computing in virtual machines[J]. *IEEE Transactions on Computers*, 2012, 61(6): 804-816.
- [11] BALSEIRO S R, LOISEAU I, RAMONET J. An ant colony algorithm hybridized with insertion heuristics for the time dependent vehicle routing problem with time windows[J]. *Computers and Operations Research*, 2011, 38(6): 954-966.
- [12] GHAFURIAN S, JAVADIAN N. An ant colony algorithm for solving fixed destination multi-depot multiple traveling salesman problems[J]. *Applied Soft Computing*, 2011, 11(1): 1256-1262.
- [13] of Ethernet with MOOSE[EB/OL]. [2013-10-10]. <http://tools.ietf.org/pdf/draft-male-armd-moose-00.pdf>.
- [7] WANG J, ZHAO W, YANG S, et al. FSDM: floodless service discovery model based on software-defined network[C]// *Proceedings of the 2013 IEEE International Conference on Communications Workshops*. Piscataway: IEEE Press, 2013: 230-234.
- [8] GREENBERG A, HIALMTYSSON G, MALTZ D, et al. A clean slate 4D approach to network control and management[J]. *ACM SIGCOMM Computer Communication Review*, 2005, 35(5): 41-54.
- [9] YAN H, MALTZ D A, NG T S E, et al. Tesseract: a 4D network control plane[C]// *Proceedings of the 4th USENIX Symposium on Networked Systems Design and Implementation*. Berkeley: USENIX, 2007: 369-382.
- [10] FEAMSTER N, REXFORD J, ZEGURA E. The road to SDN: an intellectual history of programmable networks[J]. *ACM SIGCOMM Computer Communication Review*, 2014, 44(2): 87-98.
- [11] GUDE N, KOPONEN T, PETTIT J, et al. NOX: towards an operating system for networks[J]. *ACM SIGCOMM Computer Communication Review*, 2008, 38(3): 105-110.
- [12] MCKEOWN N, ANDERSON T, BALAKRISHNAN H, et al. OpenFlow: enabling innovation in campus networks[J]. *ACM SIGCOMM Computer Communication Review*, 2008, 38(2): 69-74.
- [13] LANTZ B, HELLER B, MCKEOWN N. A network in a laptop: rapid prototyping for software-defined networks[C]// *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*. New York: ACM Press, 2010: Article No. 19.

(上接第 3191 页)

同时设计有效时间和过期时间,在确保主机信息表的时效性和数据一致性前提下,尽量减轻控制器的负载。但是,本文方案仅考虑对 ARP 广播包的处理,其他的如 DHCP、网络时间协议(Network Time Protocol, NTP)以及网络基本输入/输出系统(Network Basic Input/Output System, NetBIOS)等依赖广播的协议亟待进一步研究。并且,加入对安全因素的考虑(如预防 ARP 欺骗)也是未来工作的研究重点。

参考文献:

- [1] ELMELEEGY K, COX A L. EtherProxy: Scaling Ethernet by suppressing broadcast traffic[C]// *INFOCOM 2009: Proceedings of the 28th IEEE International Conference on Computer Communications*. Piscataway: IEEE Press, 2009: 1584-1592.
- [2] SUN X, WANG Z. An efficient and scalable metro-Ethernet architecture[J]. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 2009, 3(4): 25-42.
- [3] PLUMMER D C. RFC 826, an Ethernet address resolution protocol: converting network protocol addresses to 48-bit Ethernet address for transmission on Ethernet hardware[S]. Reston: IEIF, 1982.
- [4] KIM C, MATTHEW C, JENNIFER R. Floodless in SEATTLE: a scalable Ethernet architecture for large enterprises[J]. *ACM SIGCOMM Computer Communication Review*, 2008, 38(4): 3-14.
- [5] TOUCH J, PERLMAN R. RFC 5556, TRansparent Interconnection of Lots of Links (TRILL): problem and applicability statement[S]. Reston: IEIF, 2009.
- [6] SCOTT M, MOORE A, CROWCROFT J. Addressing the scalability