

适用于 P2P 大型多人在线游戏的包聚合机制

石祥滨, 姜佩丽, 杜玲, 毕静, 夏秀峰

(沈阳航空工业学院计算机学院, 沈阳 110136)

摘要: 提出一种适合于 P2P 大型多人在线游戏的包聚合机制。该机制基于游戏交互阈值和聚合阈值对游戏包进行聚合, 减少数据包的发送次数, 降低由数据包头部引起的开销, 同时使用平均聚合度和平均聚合延迟对该机制进行评估。实验结果表明, 该机制能够节省带宽, 满足游戏对实时性的要求。

关键词: 大型多人在线游戏; 包聚合; 游戏交互阈值

Packet Aggregation Mechanism for P2P Massively Multiplayer Online Game

SHI Xiang-bin, LOU Pei-li, DU Ling, BI Jing, XIA Xiu-feng

(Dept. of Computer, Shenyang Institute of Aeronautical Engineering, Shenyang 110136)

【Abstract】 A packet aggregation mechanism for P2P Massively Multiplay Online Game(MMOG) is proposed, which aggregates game packets into a single packet based on game interactivity threshold and aggregation threshold. The number of transmissions for data packets and the load caused by the header of packets are reduced. Moreover, an analysis method is developed to evaluate this aggregation mechanism in terms of the average aggregation degree and aggregation latency. Experimental results show this aggregation mechanism saves bandwidth and satisfies the real-time requirement of P2P MMOG.

【Key words】 Massively Multiplayer Online Game(MMOG); packet aggregation; Game Interactivity Threshold(GIT)

1 概述

在大型多人在线游戏(Massively Multiplayer Online Game, MMOG)中游戏状态的变化是以数据包形式通过网络传送给同一兴趣域内的玩家。对几款大型网络游戏的跟踪调查^[1-2]发现, 和有效负载相比, 由客户端产生的数据包极小, 包头的开销很大。因此, 本文考虑用聚合技术合并这些小的更新数据包, 以减少由数据包头部带来的开销。

一些学者对聚合技术进行了大量研究。Badia等人^[3]利用空间、时间和语义的相关性, 把要处理的消息从不同源端聚合在一起。Sangheon等人^[4]提出基于应用感知的数据聚合机制, 根据无线传感器网络产生数据包的实时特性, 并将其分为实时包和非实时包, 据此对数据进行聚合。Jouni等人提出基于时间花费和基于数量的 2 种包聚合方法, 前者的缺点是在最坏情况下, 仅有 0 或 1 个包被合并, 这样并不能节省带宽; 后者对发送延迟没有保证。

针对上述情况, 本文根据游戏包的特点提出一种适合于 P2P MMOG 的包聚合机制, 并运用平均聚合度和平均聚合延迟对其进行评估。

2 游戏数据包特点分析

Kim J 等人使用“tcpdump”软件测量工具连续 8 天跟踪 2D 网络游戏 Lineage, 搜集了 280 GB 的相关数据。分析结果表明, 由客户端发往服务器端的数据包的平均有效负载为 9.03 Byte。在跟踪 Lineage 后发现, 大型多人在线角色扮演游戏(Massively Multiplayer Online Role Play Game, MMORPG)的网络流量由 2 种包组成: 上行数据包(由客户端产生)和下行数据包(由服务器产生)。除去 TCP/IP 的头部后,

上行数据包的平均大小为 19.06 Byte; 下行数据包的平均大小为 318.39 Byte。在跟踪一款基于 TCP 的中等大小的商业性 MMORPG——“神州在线”的 135.6 亿个包后发现, 客户端产生的包极小, 98% 的包的有效负载小于 32 Byte; 服务器产生的包的平均有效负载为 114 Byte。

可以看出, 游戏世界中产生的数据包较小, 而 UDP/IP 和 TCP/IP 的头部分别占 28 Byte 和 40 Byte。与有效负载相比, TCP/IP 和 UDP/IP 头部的开销很大。

3 包聚合机制

包聚合是指将若干个包合并成单个数据包(称其为聚合包)。聚合包中能够容纳的游戏包的最大数目称为聚合阈值。包聚合减少了由数据包头部引起的开销以及数据包的发送次数, 节省了带宽。此外, 考虑到游戏的实时性要求, 引入游戏交互阈值(Game Interactivity Threshold, GIT)来保证游戏的交互性。GIT 是游戏在不失去交互性的同时能够容忍延迟的最大值, 它与游戏类型有关, 其值一般在 100 ms~200 ms 之间。

根据游戏数据包的特点, 本文基于游戏交互阈值和聚合阈值对游戏包进行聚合。当满足下列条件之一就进行数据包聚合: (1) 游戏包的数量达到一个固定值 D (即聚合阈值为 D);

基金项目: 辽宁省教育厅攻关计划基金资助项目(2004D1116); 辽宁省自然科学基金资助项目(20052007)

作者简介: 石祥滨(1963 -), 男, 教授、博士, 主研方向: 分布式系统, 网络游戏, 虚拟现实, 协同工作; 姜佩丽, 硕士研究生; 杜玲、毕静, 助教、硕士; 夏秀峰, 教授、博士

收稿日期: 2008-02-20 **E-mail:** sxb@syiae.edu.cn

(2)满足一定的时间间隔期限(将 GIT 的值定义为时间间隔)。图 1 解释了发生包聚合的 2 种情况。

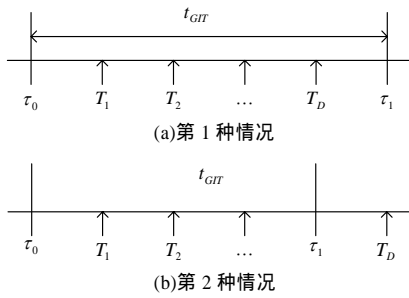


图 1 聚合的 2 种情况

其中, t_{GIT} 分别把时间点 τ_0 、 τ_1 作为起点和终点; $[\tau_0, \tau_1]$ 表示最大时间间隔 t_{GIT} ; T_i 表示游戏包在第 i 个时间点到达。图 1(a)中在 t_{GIT} 到达期限前, 已有 D 个游戏包到达, 因此在 T_D 时刻将这 D 个游戏包聚合成一个大的聚合包。图 1(b)说明了在 t_{GIT} 这段时间间隔内没有达到 D 个游戏包, 但已到达时间期限, 因此, 在 τ_1 时刻将现有的游戏包进行聚合。

本文提出的聚合机制由 3 个模块组成: 队列模块, 聚合模块和传输模块。游戏世界中产生的消息对实时性要求以及重要程度都有所不同, 假设每条消息有 2 个限定值:

(1)urgency: 表示消息的紧急程度。urgency 越高, 要求的延迟越小, 游戏包越需要被立即传送。

(2)relevance: 表示事件的重要程度。将其映射到重传次数, 以保证消息的可靠传输。

令每个玩家维护一个大小为 D 的队列, 用于放置玩家自身状态改变所产生的数据包。此队列具有优先级, 把事件的 urgency 作为第 1 个标准, 将要传送事件的最近截止日期作为排序的第 2 个标准。根据事件的紧急程度确定传送的截止日期, 并将它插入包队列。假设队列最多能聚集 D 个游戏包且不违背游戏对延迟的要求, 游戏包先在队列模块中进行排队, 满足下面条件之一: (1)当队列已满, 也就是队列规模为 D ; (2)达到固定的时间间隔 GIT, 这些包就被传送到聚合模块进行聚合。在聚合模块中, 这些游戏包被聚合成单个聚合包。聚合包里的游戏包数量与游戏交互阈值 GIT 以及聚合阈值 D 有关。聚合队列中聚集的游戏包数量为 0 个或小于等于 D 。因此, 聚合包的长度 (L_A) 的取值如下: $0 \leq L_A \leq D \cdot L$ 。其中, L 是单个游戏包的长度, 且 $L_A \leq$ MTU 的大小。这里的 MTU 是指 Maximum Transmission Unit, 即最大传输单元, 即能够通过物理网络传输的最大包的数据量, 单位是 Byte。不同网络的 MTU 不同。

对以太网而言, MTU 是 1 500 Byte。在聚合机制中, D 是一个关键参数, 该值的设定应考虑底层最大传输单元 MTU 的长度, 因为它影响延迟和带宽的消耗。当 D 增加时, 包聚合延迟变大, 反之带宽的消耗增加。最后, 聚合包通过传输模块广播给兴趣域内的其他玩家以及协调者节点。在传输时使用哪种传输协议(TCP 或 UDP)要根据具体的游戏类型来确定。对于实时策略类游戏最好使用 TCP, 对于动作类游戏应使用 UDP, 对于角色扮演类游戏最好使用 UDP, 而慢节奏的角色扮演游戏通常使用 TCP。

本文使用平均聚合度和平均聚合延迟来评估包聚合机制。聚合度是指在一个聚合包中聚集到的游戏数据包的数量, 它由聚合阈值和游戏交互阈值决定。聚合延迟被定义为传输

2 个连续聚合包的平均时间间隔。根据对网络流量的分析可以作出如下假设: 在某一时间段内游戏包的产生过程服从泊松分布, 而游戏包到达时间间隔服从指数分布, 由此可计算出平均聚合度以及平均聚合延迟, 计算过程如下:

$$E(N|t_{GIT}) = Pr(n = D|t_{GIT}) \cdot D + \sum_{i=0}^{D-1} Pr(n = i|t_{GIT}) \cdot (i+1) \quad (1)$$

式(1)表示在给定的 t_{GIT} 时间段内游戏包的平均聚合度。式(1)右边的第 1 项表示图 1(a)中的情况, 第 2 项表示图 1(b)中的情况。 n 是在 t_{GIT} 时间段内到达的游戏包的数量。假设游戏包到达的过程遵循到达率为 λ_1 的泊松分布函数:

$$Pr(n = i | t_{GIT} = \tau) = \frac{e^{-\lambda_1 \tau} \cdot (\lambda_1 \tau)^i}{i!} \quad (2)$$

游戏包到达时间间隔遵循指数分布, 平均数为 λ_2 , 那么概率密度函数 $f_{t_{GIT}}(\tau) = \lambda_2 e^{-\lambda_2 \tau}$ 。因此, 整合式(1)和式(2), 在 $[0, \infty]$ 内平均聚合度 $E(N)$ 可表示为

$$E(N) = \int_0^{\infty} E(N|t_{GIT}) \cdot f_{t_{GIT}}(\tau) d\tau = \int_0^{\infty} E(N|t_{GIT}) \cdot \lambda_2 e^{-\lambda_2 \tau} d\tau \quad (3)$$

同样, 平均聚合延迟 $E(L)$ 的计算如下:

$$E(L|t_{GIT}) = Pr(n = D|t_{GIT}) \cdot D \cdot \frac{1}{\lambda_1} + \sum_{i=0}^{D-1} Pr(n = i|t_{GIT}) \cdot \frac{1}{\lambda_2} \quad (4)$$

因此,

$$E(L) = \int_0^{\infty} E(L|t_{GIT}) \cdot f_{t_{GIT}}(\tau) d\tau = \int_0^{\infty} E(L|t_{GIT}) \cdot \lambda_2 \cdot e^{-\lambda_2 \tau} \cdot d\tau \quad (5)$$

4 实验方案及结果分析

实验采用的仿真工具是 NS2。游戏包的到达过程服从泊松分布, 游戏包的到达时间间隔服从指数分布。通过调节游戏数据包间的时间间隔来控制游戏包的发送速率。当聚合阈值 D 分别为 6, 8, 10 时, 游戏包的到达速率与平均聚合度及其与平均聚合延迟的关系如图 2、图 3 所示。

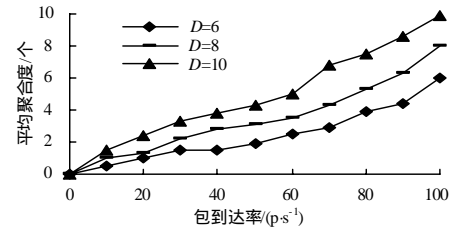


图 2 游戏包的到达率与平均聚合度的关系

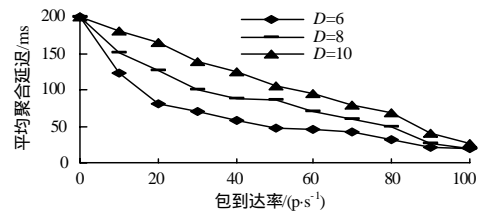


图 3 游戏包的到达率与平均聚合延迟的关系

从图 2 可以看出, 平均聚合度随着游戏包到达率的增加而增加, 且最大值不会超过聚合阈值 D 。在极端情况下, 平均聚合度为 0, 这是因为在固定时间间隔内没有任何游戏包到达。平均聚合度反映了带宽节省情况, 平均聚合度越高, 带宽节省越多。当平均聚合度为 0 时, 带宽节省为 0, 相当于没有采用包聚合机制。

从图 3 可以看出, 平均聚合延迟随着游戏包到达率的减少而增加。它小于或等于游戏交互阈值, 这说明包聚合机制所产生的延迟对游戏性能影响不大, 可以被玩家接受。

(下转第 142 页)