

# Network Analysis of Counter-strike and Starcraft

Mark Claypool, David LaPoint and Josh Winslow  
{claypool|dlapoint|jwinslow}@wpi.edu  
Computer Science Department  
Worcester Polytechnic Institute  
100 Institute Road, Worcester, MA 01609, USA

January 31, 2002

## Abstract

Network games are becoming increasingly popular, but their traffic patterns have received little attention from the academic research community. The network traffic patterns of the multiplayer games Counter-strike and Starcraft were examined and documented. Analysis focused on bandwidth usage and packet size. The games were found to have small packets and typically low bandwidth usage, but traffic patterns varied widely between games.

## 1 Introduction

The rapid growth in the connectivity of the Internet and the fall in computer hardware prices has fueled the growth of multiplayer gaming over the Internet. In addition to computers, the latest generation of consoles systems from Nintendo, Sony, and Sega all include network support for multiplayer game play on the Internet. The increase in Internet gaming is also a result of change in developer focus from single player games to multiplayer games.

Despite growth of use on the Internet, network games typically do not use traditional Internet transport protocols. The majority of Internet traffic runs on top of TCP, which guarantees sequential, in order arrival of all data by retransmitting lost packets. However, the additional time required for retransmissions and the window-based data rate imposed by TCP can be unacceptable to network games since interactive games often have strict delay constraints [Arm01]. Instead of TCP, network games often use UDP since UDP packets are not guaranteed to reach their destinations, and thus do not require a mechanism for retransmissions. Unfortunately, UDP does not provide any built-in congestion control, presenting the risk of congestion collapse as the fraction of unresponsive UDP traffic increases [FF99].

Despite the very real burden that network games may impose on network bandwidth and the massive growth and large user base, issues related to the effects of these games on network congestion have been largely neglected in both academic and industry publications. Industry articles are more concerned with the management aspects of game development or focus on latency and maximum bandwidth issues only [Lin99, BT01, Ng97], whereas attention to detail like minimizing network load suffer due to the lack of economic pressure. Network gaming has not been a traditional research field in academia as network games are viewed primarily as a diversion for students, rather than a practical problem for computer scientists.

The goal of this research is to analyze the network traffic of two of the most popular games on the Internet. We chose two games from the most popular genres, First Person Shooters and Real Time Strategies (see

Section 2.1), with the intent of representing a larger range of data than a single genre would likely provide. We used a network sniffer on both game servers and game clients participating in real Internet games in order to capture realistic network traffic. We analyze the distribution of packet sizes, differences between client and server bandwidths and impact of number of players on network traffic.

## 2 Approach

We employed the following methodology: select network games for study (Section 2.1), run the selected games in an instrumented environment (Section 2.2), and analyze network traffic the games generate (Section 3).

### 2.1 Selecting Games

Broadly, the most popular categories of real-time network games are *First Person Shooters* (FPS) and *Massively Multiplayer Online Role Playing* games (MMORP), followed closely by *Real Time Strategy* (RTS) games.

FPS games, first made popular by Doom, have the player view the world through the perspective of a character (the first person) and have the player move around slaying monsters and other players, with an amalgamation of ranged weaponry (the shooter). On an average night, there are well over 10,000 servers Half-Life<sup>1</sup> games supporting over 40,000 gamers.<sup>2</sup> Other FPS games support slightly smaller user populations.<sup>3</sup>

MMORP games, first made popular by Ultima Online's release in 1996,<sup>4</sup> are similar to graphical Multi-User Dungeons, first pioneered in the 70s.<sup>5</sup> MMORP games provide a mechanism for character advancement, large lands to travel across, and other players with whom to interact. The three biggest MMORP games, Asheron's Call<sup>6</sup>, Ultima Online<sup>7</sup>, and Everquest<sup>8</sup>, claim to have nearly 1 million subscribers combined.

RTS games, first made popular by Dune 2<sup>9</sup>, are generally characterized by resource collection, unit construction, and battles that consist of large numbers of animated soldiers going through a repetitive, animated attack. All of these actions happen in real-time, unlike earlier strategy games such as Civilization<sup>10</sup>, in which the player could take as much time as needed to plan the next turn. Since Dune 2, there have been several more RTS games released, one of the more popular being Starcraft<sup>11</sup>. Currently, the typical number RTS fans playing Starcraft on an average night numbers about 20,000 players.

We selected Starcraft, a RTS game, and Counter Strike, a FPS game, because they were familiar, best selling games in different genres. While we wanted to study Asheron's Call, a MMORP game, the time it took to setup and run our experiments precluded studying another genre. We leave that as future work.

---

<sup>1</sup><http://half-life.sierra.com/>

<sup>2</sup>Average of 3 randomly selected nights between 10/2/01 and 12/15/01

<sup>3</sup>Tribes 2, based on an average over 3 randomly selected nights between 10/16/01 and 12/15/01

<sup>4</sup><http://www.uo.com>

<sup>5</sup><http://www.legendmud.org/raph/gaming/book.html>

<sup>6</sup><http://www.zone.com/asheronscall/>, averages 12,000 players per night

<sup>7</sup><http://www.uo.com>, over 300,000 total subscribers

<sup>8</sup><http://everquest.station.sony.com/>, over 410,000 total subscribers

<sup>9</sup><http://www.dune2k.com/duniverse/dune2/>

<sup>10</sup><http://www.civ3.com>, the second sequel of Civilization

<sup>11</sup><http://www.blizzard.com/worlds-starcraft.shtml>

### 2.1.1 The Starcraft Game Environment

Starcraft is a real-time strategy game that has players construct buildings and fighting units, and issue commands that cause the units to move, engage enemy units, and similar tasks. Every game is played on one of many possible maps, either provided with the game or custom built by users. There are three races from which a player can choose, and each has a balanced set of advantages and disadvantages over the others. There are a number of ways in which players can be competitively grouped. In a free-for-all game, all players vie to have the last remaining army on the map. Players can also team up against each other and/or against AI scripted “computer” players in myriad ways.

For our experiments, all games were structured so that there were two teams of equivalent sizes: 2 vs. 2, 3 vs. 3, 4 vs. 4, and 8 vs. 8 players. The local player, analyzed in detail in Section 3, played as the same race in each game, and employed the same building strategy throughout. The games were played using Starcraft: Brood War, version 1.7; the local player logged on to Battle.net using the USEAST gateway, and created the game sessions; each game type was top players vs. bottom players; and the same map, called Big Game Hunters found in the maps/broodwar/webmaps directory from where the game was installed, was used for each game.

### 2.1.2 The Counter-Strike Game Environment

Counter-strike<sup>12</sup> is a modification to Half-Life that is distributed free over the Internet for owners of Half-Life, or as a retail product in most game stores. Counter-strike puts the players in the role of either a terrorist attempting to hold hostages, blow up landmarks, or assassinate a VIP or a counter-terrorist agent trying to thwart the terrorists. To play, each player must connect to a server, usually located on an end-host on the Internet. When one or more other players join the same server, a game begins.

All Counter-strike games are played on a map, each of which has its own set of objectives. Most objectives have the counter-terrorists attempting to rescue a set of hostages from close to where the terrorists start, or the terrorists attempting to plant a bomb close to where the counter-terrorists start. The map is played several times, with each time being called a round, lasting several minutes. Each round ends either when the victory conditions are met, time runs out, or when one team has been totally eliminated. At the start of each round, both sides are allowed to buy weapons and ammunition with the money they earned from the previous rounds. The better a team did in the previous round, the more money they have to spend. Once each team is equipped, they attempt to wipe out the other team with their weaponry or complete the objective, although the former outcome ends far more rounds than the latter.

Controlling the environment in a real Counter-strike game is difficult because players can join and leave as they please. We chose to analyze several different maps to see the effect the campaign has on network traffic. The number of players was not controlled, as players joined and left the server in a typical Counter-Strike gaming fashion. We used Counter-strike version 1.3 on Half-life version 1.1.0.8. The maps used were from the standard install: de\_dust, de\_aztec, and cs\_assault. The Counter-strike server we connected to was located on the WPI network.

---

<sup>12</sup><http://www.counter-strike.net>

## 2.2 Running Games

We used Commview (version 2.6, build 103)<sup>13</sup> to capture all network traffic related to the games ran. Commview is a robust sniffer with the ability to filter packets, compute statistics, and generate reports periodically.

All Starcraft data was collected on the machine:

```
Intel Pentium III 800mhz processor with 100mhz FSB
512 megabytes PC-100 SDRAM
nVidia GeForce2 3d graphic accelerator with 64 megabytes of DDR SDRAM
UltraWide SCSI hard drive interface
10baseT network card connected to 608/108 kbps DSL modem
Windows 98B Operating System
```

All Counter-strike data was collected on the machine:

```
AMD Athlon 800mhz processor with 200mhz FSB
256 megabytes PC-100 SDRAM
nVidia GeForce 3d graphic accelerator with 32 megabytes of DDR SDRAM
ATA-66 hard drive interface
10baseT network card connected to WPI LAN through residence hall connection
Windows 98 v4.10.98 Operating System
```

## 3 Results

This section presents some of the results on Starcraft packet size and bandwidth usage (Section 3.1) and Counter-Strike packet size and bandwidth usage (Section 3.2).

### 3.1 Starcraft

Figure 3.1 (LEFT) depicts a histogram of the packet-sizes sent for 2 player, 4 player, 6 player and 8 player Starcraft games. The number of players does not have a noticeable effect on the packet sizes. The majority of the packets are small, with 99% of all packets being less than 150 bytes. 75% of the packets are 132 bytes, with clusters of packets at 118, 120 and 140 bytes.

Figure 3.1 (RIGHT) depicts the bandwidth sent by each player for 2 player, 4 player, 6 player and 8 player Starcraft games. The amount of bandwidth sent is linear with the number of game players. A 2-player game sends around 650 bytes/second and each pair of additional players adds about 1500 bytes/second of data. Also, as the number of players increases, so does the variance in bandwidth sent.

### 3.2 Counter-strike

While a Starcraft game has very little deviation in terms of packet size or bandwidth used throughout the run of one game, Counter-strike, especially from the viewpoint of a server side, has a non-uniform, but distinct network pattern.

Figure 2 (LEFT) depicts the bandwidth sent for a Counter-strike game using the `cs_assult` map for the first 20 minutes and then switching to the `de_aztec` map. The average bandwidth sent is 2694 bytes/second.

---

<sup>13</sup><http://www.tamos.com/products/commview/>

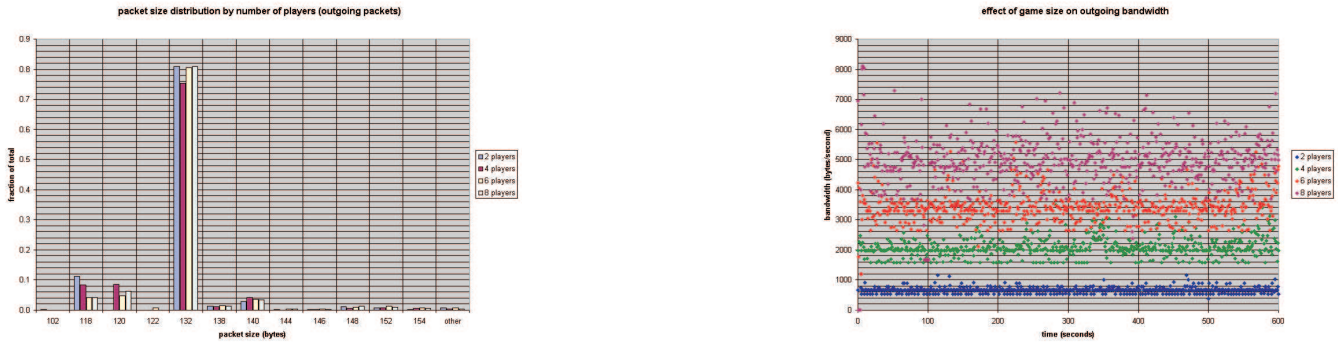


Figure 1: LEFT: Starcraft Packet Size Distribution Sent for Different Game Sizes. RIGHT: Starcraft Bandwidth Sent for Different Game Sizes

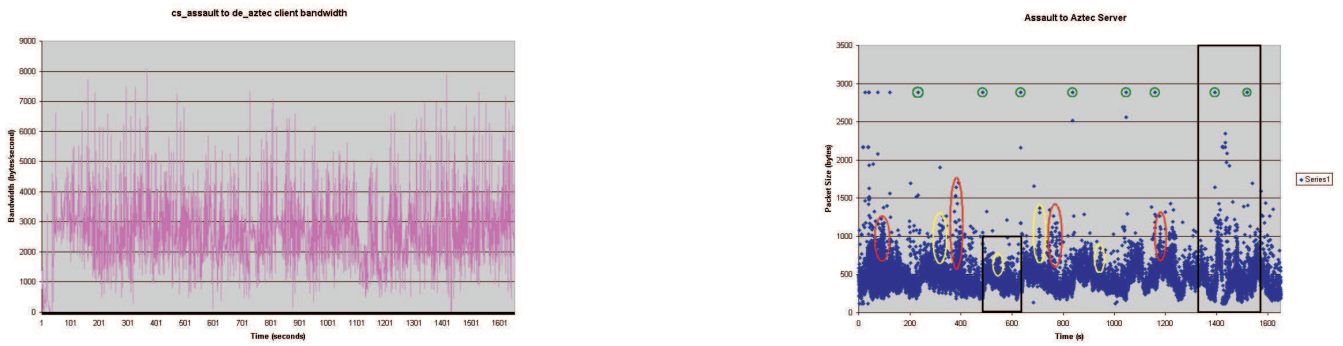


Figure 2: LEFT: Counter-strike Bandwidth Sent by Client. RIGHT: Counter-strike Packet Sizes Sent by Server

There is considerable variation in the bandwidth per second, with a noticeable drop in bandwidth at 1100 seconds when there is a map change.

Figure 2 (RIGHT) depicts the packet sizes sent over time for a Counter-strike game using the `cs_assault` map for the first 23 minutes and then switching to the `de_aztec` map. Rounds can be distinguished by the slow decline, caused by players dying, in the packet sizes sent from the server. For example, one round goes from approximately 500s to 650s, featured above in the first small box. The mean packet size is 465 bytes. It seems likely that the large packets of nearly 3000 bytes, circled, are round initialization or round termination packets, as they strongly correlate with the end of the decline in packet sizes. The ovals correlate with large firefights within the game. The break in the graph around 1400 seconds, indicated by the second large box, is where the map was changed to `de_aztec`.

Figure 3 depicts the bandwidth sent by the server for a Counter-strike game using the `cs_assault` map for the first 20 minutes and then switching to the `de_aztec` map. The average bandwidth is 5871 bytes/second, with considerable variation in the bandwidth per second. The map change occurs at about 1110 seconds, depicted by the drop in bandwidth.

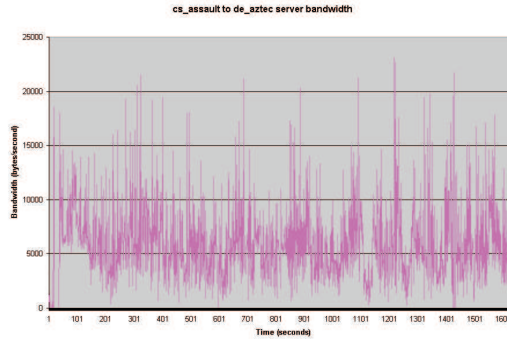


Figure 3: Counter-strike Bandwidth Sent by Server

## 4 Summary

The network traffic generated by Starcraft and Counter-strike look very different. There is a significant degree of variation in the packet sizes and bandwidth used in Counter-strike games, in contrast to Starcraft in which the differences between sizes of packets sent are barely distinguishable. For larger games, Starcraft does have more packets transmitted when the bandwidth requirements increase. The Counter-strike client also follows this model, although the packet sizes are more variable than those in Starcraft. The Counter-strike server, on the other hand, increases the size of the packets when it needs to send more data to the client. These differences are not visible when viewing a bandwidth graph but are important to note due to their possible effects on network congestion.

Starcraft and Counter-strike are also different in the bandwidth they consume over time. Starcraft's bandwidth consumption varies very little over the course of a game regardless of events occurring within the game. Counter-strike, however, has a distinct cyclic pattern in bandwidth, which vary over time as players get killed, and have a marked correlation to game events. Overall, the amount of bandwidth consumed by a Starcraft player is comparable to bandwidth consumed by a Counter-strike client. A 6-player game of Starcraft has each client send between 3000 and 3500 bytes/second, and a Counter-strike client connected to a mostly-full server typically sends a little over 3200 bytes/second.

## References

- [Arm01] Grenville Armitage. Lag Over 150 Milliseconds is Unacceptable, May 2001. [Online] at <http://members.home.net/garmitage/things/quake3-latency-051701.html>.
- [BT01] Paul Bettner and Mark Terrano. 1500 Archers on a 28.8: Network Programming in Age of Empires and Beyond. *Gamasutra*, March 2001. [Online] at [http://www.gamasutra.com/features/20010322/terrano\\_02.htm](http://www.gamasutra.com/features/20010322/terrano_02.htm).
- [FF99] Sally Floyd and Kevin Fall. Promoting the Use of End-to-End Congestion Control in the Internet. *IEEE/ACM Transactions on Networking*, February 1999.
- [Lin99] Peter Lincroft. The Internet Sucks: Or, What I Learned Coding X-Wing vs. Tie Fighter. *Gamasutra*, September 1999. [Online] at [http://www.gamasutra.com/features/19990903/lincroft\\_01.htm](http://www.gamasutra.com/features/19990903/lincroft_01.htm).
- [Ng97] Yu-Shen Ng. Designing Fast-Action Games for the Internet. *Gamasutra*, September 1997. [Online] at [http://www.gamasutra.com/features/19970905/ng\\_01.htm](http://www.gamasutra.com/features/19970905/ng_01.htm).