

# Relatório Trabalho Redes

Alunos:

Higor Gabriel

Vitor Gabriel

Mat.: 0070308

Mat.: 0071054

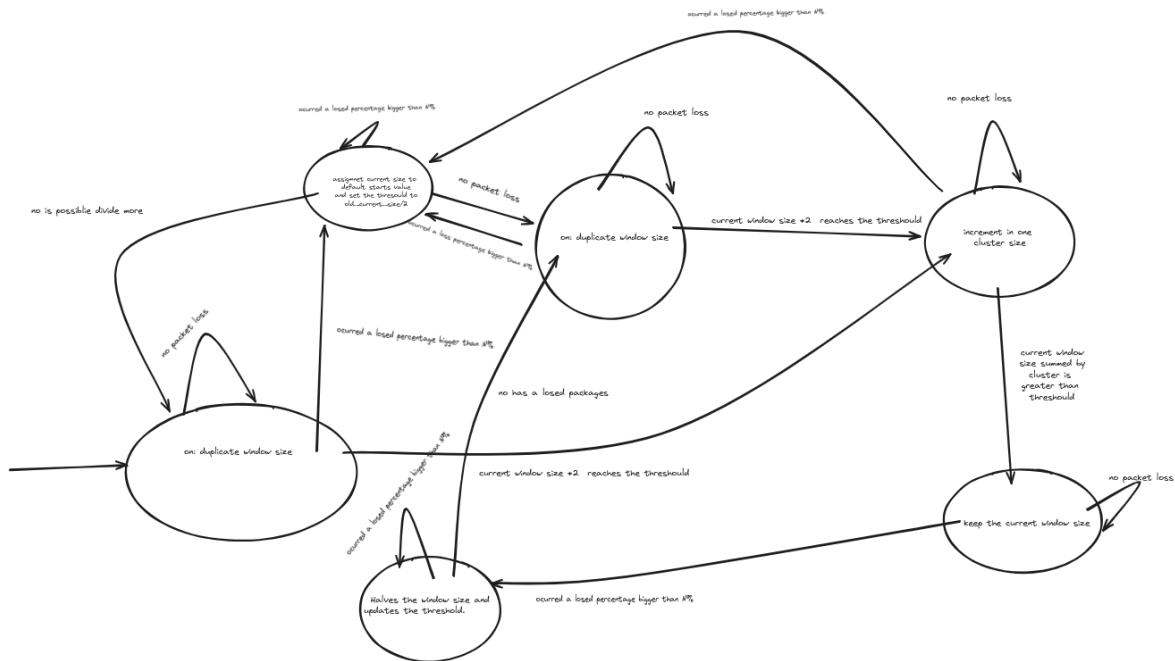
## Arquitetura:

Foi implementado um sistema de controle de janela que foi inspirada pelos princípios do protocolo TCP, especificamente nas estratégias de Slow Start e AIMD (Additive Increase/Multiplicative Decrease). Assim como o TCP usa Slow Start para aumentar gradualmente a janela de congestionamento até detectar uma perda de pacotes, a classe `WindowHandler` começa com um tamanho de janela menor e o duplica em estados iniciais, ajustando-se conforme as condições de perda. A diferença principal é que, enquanto o TCP realiza uma redução multiplicativa da janela em caso de perda (AIMD), o código implementa uma abordagem mais controlada e adaptativa, com transições entre vários estados que ajustam o tamanho da janela de forma incremental ou através de recuperação lenta, ao invés de uma redução abrupta. Isso permite uma resposta mais fina às condições da rede, potencialmente resultando em uma transmissão mais eficiente em cenários específicos.

A máquina de estados usa como limite superior 150% do byterate do vídeo, isso significa que ela nunca vai ultrapassar essa faixa de transmissão, como limite inferior usamos 20% do byterate. Para fazer as transições em casos de perdas, somente a fazemos se a perda de bytes foi maior do que uma porcentagem definida na variável de ambiente

`AT_MOST_LOSS_PERCENTAGE`.

A máquina de estados para esta implementação é a seguinte:



O sistema é genérico, as métricas utilizadas (neste caso quando há perda) podem ser modificadas para considerar outras coisas e aí sim decidir como continuar mandando.

Existe uma classe chamada `StreamLayer` no servidor, cuja principal função é fornecer métodos para sincronização entre a thread de controle (responsável por receber os comandos) e a thread de stream (a qual é criada automaticamente para manipular o vídeo toda vez que a classe é inicializada). Portanto, toda a lógica de sincronização de threads está encapsulada nesta classe. Assim, a classe `ApiSession` processa e faz o parse dos comandos e, quando precisa interagir com algum recurso compartilhado com a thread de stream (como o arquivo do vídeo, por exemplo), utiliza os métodos fornecidos pela classe `StreamLayer`.

Quando o cliente precisa de mais vídeo para enviar para o pipe, ele envia uma requisição com o comando `get_video_frame`, que é atendida de forma assíncrona. O servidor então obtém o tamanho da janela atual do TAD da máquina de estados e coloca a solicitação em uma fila na classe `StreamLayer`. Por outro lado, comandos como `seek`, `pause`, `stop`, `seek_forward` e `seek_backward` são executados de forma síncrona e não passam pela fila; eles são processados imediatamente no momento em que são recebidos.

## Utilização dos canais TCP/UDP

É utilizado no sistema dois canais de dados, sendo eles: TCP para troca de comandos e mensagens, e o canal UDP para o envio e recebimento dos dados do vídeo que está sendo transmitido, o que garante confiabilidade na entrega dos comandos pela parte do TCP e uma boa vasão nos dados do vídeo pelo UDP.

## Pontos Fortes e Fracos

### Pontos Fortes:

1. **Adaptabilidade:** O sistema foi projetado para ser altamente versátil e adaptável. Ele permite ajustes no controle da janela de transmissão, o que pode ser facilmente modificado para se adequar a diferentes condições de rede ou requisitos específicos.
2. **Implementação Inspirada em TCP:** A arquitetura baseada nos princípios de Slow Start e AIMD oferece um modelo bem fundamentado para controle de congestionamento, aproveitando estratégias já comprovadas no protocolo TCP. Isso permite ao sistema manter um equilíbrio entre eficiência e estabilidade na transmissão de dados.
3. **Recuperação Gradual:** Em vez de uma redução abrupta da janela de transmissão em caso de perda de pacotes, o sistema utiliza uma abordagem mais gradual e controlada.

#### **Pontos Fracos:**

1. **Dependência de Métricas Específicas:** Apesar da adaptabilidade, as métricas atualmente utilizadas para definir as transições de estados e ajustes da janela não foram totalmente otimizadas. Isso pode levar a uma má utilização do potencial do sistema ou a uma resposta inadequada em certos cenários.

## **Análise de desempenho:**

Como a máquina de estados implementada tem o intuito de se adaptar a quaisquer condições de rede ou desempenho do cliente, é normal que ela sempre vá de forma gananciosa tentar transmissões altas, dito isso, segue uma análise de vazão e perda para diferentes vídeos:

**Sintel2010.1080p.mkv:**

```
> > > ▾ TERMINAL
stream-server-1 | INFO:root:update window size to 1929216
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 1929216, bytes lost: 0
stream-server-1 | INFO:root:update window size to 1933312
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 1933312, bytes lost: 0
stream-server-1 | INFO:root:update window size to 1937408
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 1937408, bytes lost: 0
stream-server-1 | INFO:root:update window size to 1941504
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 1941504, bytes lost: 0
stream-server-1 | INFO:root:update window size to 1945600
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 184464, bytes lost: 1761136
stream-server-1 | INFO:root:update window size to 98304
stream-server-1 | INFO:root:current state: 1
stream-server-1 | INFO:root:throughput: 98304, bytes lost: 0
stream-server-1 | INFO:root:update window size to 196608
stream-server-1 | INFO:root:current state: 2
stream-server-1 | INFO:root:throughput: 196608, bytes lost: 0
You, 11 hours ago
```

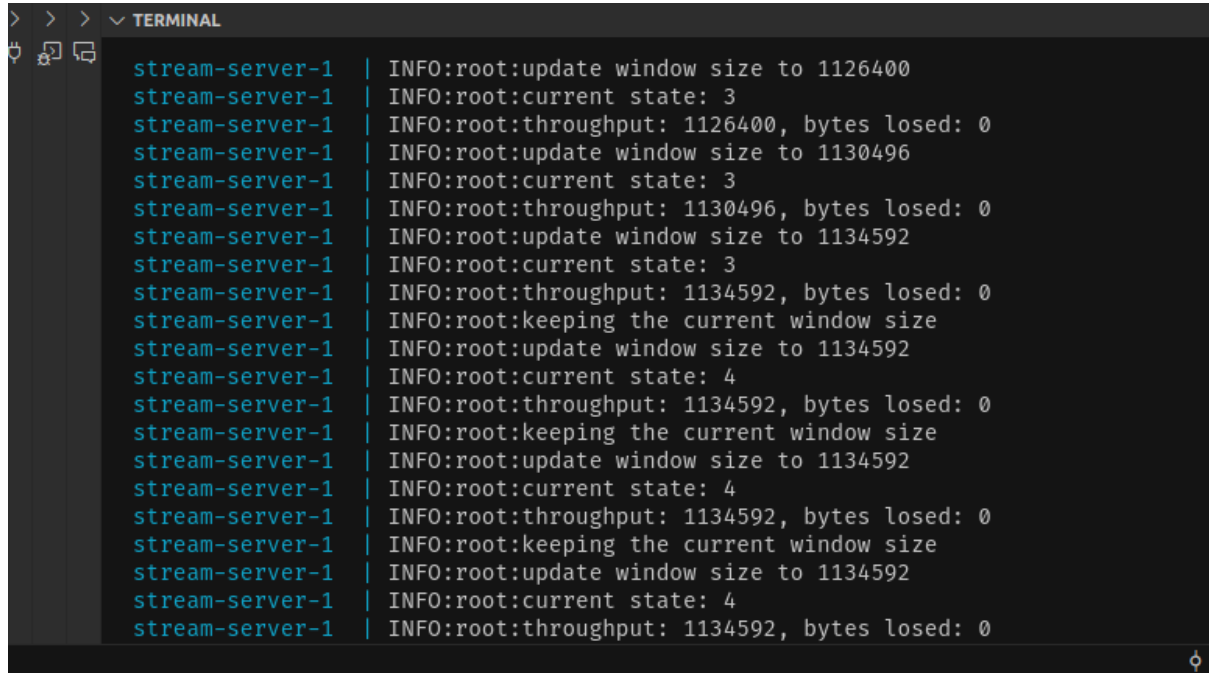
O seguinte print mostra as transições da máquina para o vídeo, podemos perceber que ela estava constantemente no estado 3 incrementando o tamanho da janela com uma alta vazão e zero perdas, mas ao ocorrer a primeira perda, onde o cliente recebeu somente 184464 bytes e perderam-se 1761136, ela foi para o estado 1, onde ela reseta novamente para o limite inferior, e seta o limite superior como a metade a última janela, após isso, depois de um certo tempo, ela conseguiu chegar a seguinte vazão estável para este vídeo:

```
> > > ▾ TERMINAL
stream-server-1 | INFO:root:throughput: 372736, bytes lost: 0
stream-server-1 | INFO:root:update window size to 376832
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 376832, bytes lost: 0
stream-server-1 | INFO:root:update window size to 380928
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 380928, bytes lost: 0
stream-server-1 | INFO:root:update window size to 385024
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 385024, bytes lost: 0
stream-server-1 | INFO:root:update window size to 389120
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 389120, bytes lost: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 389120
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 389120, bytes lost: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 389120
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 389120, bytes lost: 0
You, 11 hours ago
```

O estado 4 é o estado onde a máquina mantém a menos que ocorra perdas

## Sintel.2010.720p.mkv:

Para este vídeo, com byterate menor (devido a qualidade menor), a máquina de estados conseguiu avançar do primeiro ao 4 estado sem perdas, pois este byterate do video não causava problemas na rede ou no cliente



```
> > > ▼ TERMINAL
stream-server-1 | INFO:root:update window size to 1126400
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 1126400, bytes losed: 0
stream-server-1 | INFO:root:update window size to 1130496
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 1130496, bytes losed: 0
stream-server-1 | INFO:root:update window size to 1134592
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 1134592, bytes losed: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 1134592
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 1134592, bytes losed: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 1134592
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 1134592, bytes losed: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 1134592
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 1134592, bytes losed: 0
```

Pouco tempo após, houve uma pequena perda, devido à alguma variação na rede, mas devido a essa perda, a máquina de estado entrou no estado 5 (estado de recuperação rápida), e conseguiu um novo tamanho de janela supostamente adequado (valor de janela onde as perdas foram baixas).

```
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 1134592
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 1134592, bytes losed: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 1134592
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 1134592, bytes losed: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 1134592
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 1134592, bytes losed: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 1134592
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 1134592, bytes losed: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 1134592
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 1134592, bytes losed: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 1134592
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 187392, bytes losed: 947200
stream-server-1 | INFO:root:update window size to 1126400
stream-server-1 | INFO:root:current state: 5
stream-server-1 | INFO:root:throughput: 243024, bytes losed: 883376
stream-server-1 | INFO:root:update window size to 1118208
stream-server-1 | INFO:root:current state: 5
stream-server-1 | INFO:root:throughput: 1118208, bytes losed: 0
stream-server-1 | INFO:root:update window size to 1118208
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 1118208, bytes losed: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 1118208
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 1118208, bytes losed: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 1118208
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 1118208, bytes losed: 0
```

BigBuckBunny\_640x360.m4v:

```
stream-server-1 | INFO:root:throughput: 270336, bytes lost: 0
stream-server-1 | INFO:root:update window size to 274432
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 274432, bytes lost: 0
stream-server-1 | INFO:root:update window size to 278528
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 278528, bytes lost: 0
stream-server-1 | INFO:root:update window size to 282624
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 282624, bytes lost: 0
stream-server-1 | INFO:root:update window size to 286720
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 286720, bytes lost: 0
stream-server-1 | INFO:root:update window size to 290816
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 290816, bytes lost: 0
stream-server-1 | INFO:root:update window size to 294912
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 294912, bytes lost: 0
stream-server-1 | INFO:root:update window size to 299008
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 299008, bytes lost: 0
stream-server-1 | INFO:root:update window size to 303104
stream-server-1 | INFO:root:current state: 3
stream-server-1 | INFO:root:throughput: 303104, bytes lost: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 303104
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 303104, bytes lost: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 303104
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 303104, bytes lost: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 303104
stream-server-1 | INFO:root:current state: 4
stream-server-1 | INFO:root:throughput: 303104, bytes lost: 0
stream-server-1 | INFO:root:keeping the current window size
stream-server-1 | INFO:root:update window size to 303104
stream-server-1 | INFO:root:current state: 4
```

Para esta vídeo, não houve perdas ou problemas, logo a máquina conseguiu atingir o estado 4 com um janela quase perto de 150% do byterate sem maiores problemas.