

UNIVERSIDADE FEDERAL DE ALFENAS

FLAVIANE MOURA OLIVEIRA

PRIMEIRO TRABALHO PRÁTICO DE ANÁLISE DE DESEMPENHO

ALFENAS - MG

2022

Relatório

1. Introdução

O objetivo deste trabalho foi trabalhar com o Simulador proposto em sala de aula, modificando devidamente o código fonte de modo a coletar dados a cada 100 segundos (exatos) de simulação.

Consideremos alguns parâmetros fixos na simulação:

- Taxa média de chegada: 5/segundo, ou seja, o intervalo médio entre chegadas deverá ser igual a 0,2 segundos.
- Tempo de simulação: 36000 segundos, o que seria equivalente a 10 horas de simulação.

Após definidos estes parâmetros, foi calculado matematicamente o tempo médio de serviço para os quatros cenários propostos, sendo eles:

1. 80% de taxa de ocupação: 0,16
2. 90% de taxa de ocupação: 0,18
3. 95% de taxa de ocupação: 0,19
4. 99% de taxa de ocupação: 0,198

Calculados os valores, foi executado o simulador para os quatros cenários a fim de coletar os valores finais de Ocupação, medidas de Little ($E[N]$ e $E[W]$) e Erro de Little.

2. $E[N]$

$E[N]$ é uma medida de desempenho na validação matemática, sabemos que $E[N]$ é o número médio de elementos no sistema. Para calcular tal parâmetro, precisamos manipular três variáveis, sendo elas:

- numeroEventos
- somaAreas
- tempoAnterior

```
e_n.soma_areas +=  
(tempo_decorrido - e_n.tempo_anterior) * e_n.no_eventos;
```

Cálculo $E[N]$ no Simulador em linguagem C

Posto isso, partimos então para o cálculo de $E[N]$ final, o qual obtemos após obter o valor da soma das áreas de todos os $E[N]$ coletados durante a simulação dividido pelo tempo de coleta.

```
double e_n_final = (e_n.soma_areas) / tmp_col;
```

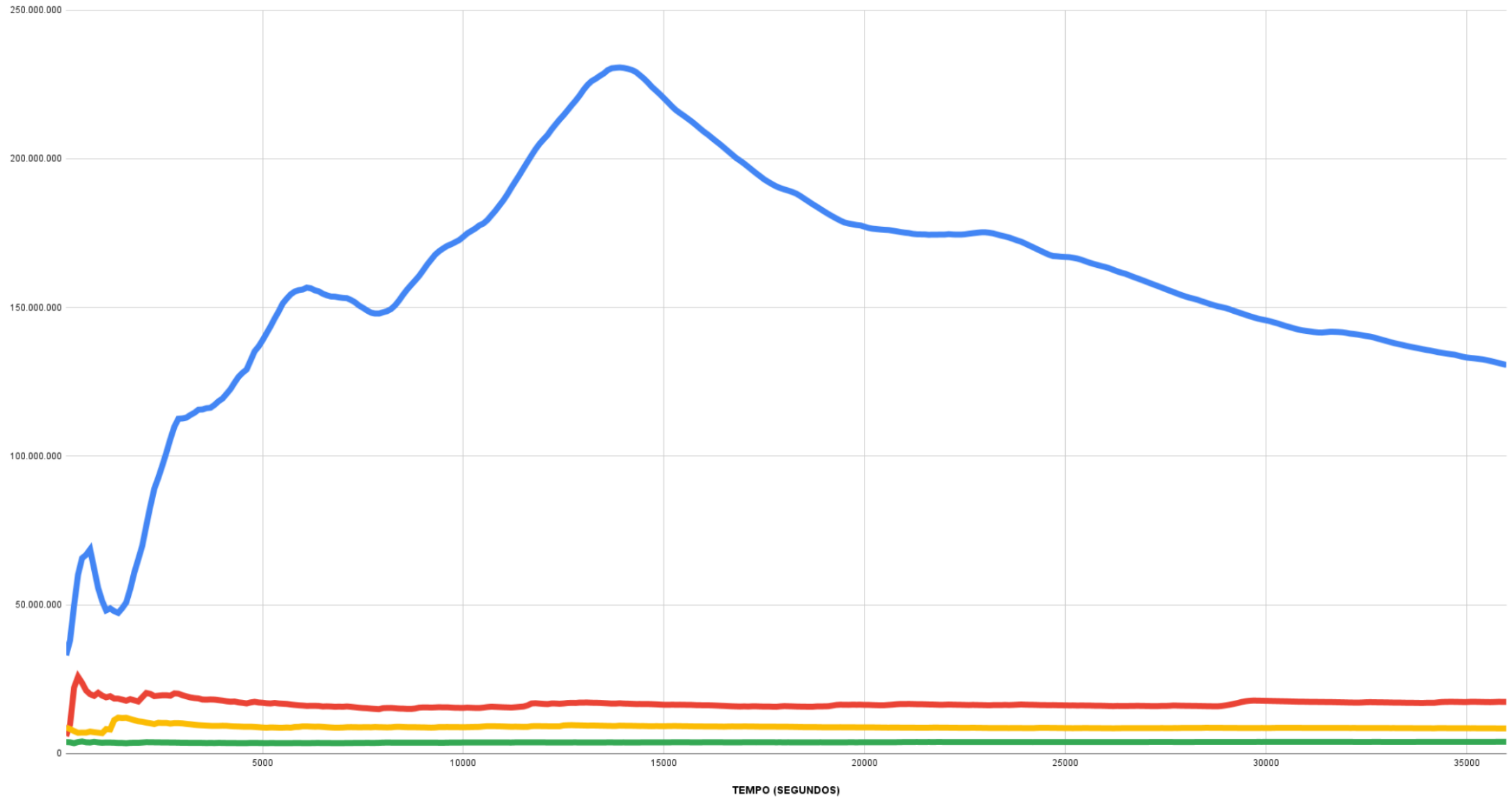
Cálculo $E[N]$ final no Simulador em linguagem C

Considerando os quatro cenários distintos, podemos ver, no gráfico abaixo, a variação de $E[N]$ a partir da alteração no valor final da taxa de ocupação. Nota-se que quanto maior a taxa de ocupação, maior será a medida de $E[N]$.

Podemos observar também que ao decorrer do tempo, o valor de $E[N]$ tende a ir estabilizando-se.

E[N] FINAL

— E[N] (99%) — E[N] (95%) — E[N] (90%) — E[N] (80%)



3. E[W]

E[W] é uma medida de desempenho na validação matemática, sabemos que E[W] é o tempo médio de espera dos elementos no sistema. Para calcular tal parâmetro, precisamos manipular duas variáveis, sendo elas:

- $\alpha(t)$: número de chegadas até determinado tempo (t)
- $\beta(t)$: número de saídas até determinado tempo (t)

Tendo estas duas variáveis, podemos então calcular o tamanho da fila em determinado instante (t), através da seguinte fórmula:

$$\alpha(t) - \beta(t)$$

Através destes dados é só então que conseguimos obter o valor de E[W].

```
e_w_chegada.soma_areas +=  
(tempo_decorrido - e_w_chegada.tempo_anterior) * e_w_chegada.no_eventos;
```

Cálculo E[W] no Simulador em linguagem C

Vale ressaltar que obtemos o valor de E[W] final através do cálculo da área entre a diferença do tempo médio de chegadas pelo tempo médio de saídas.

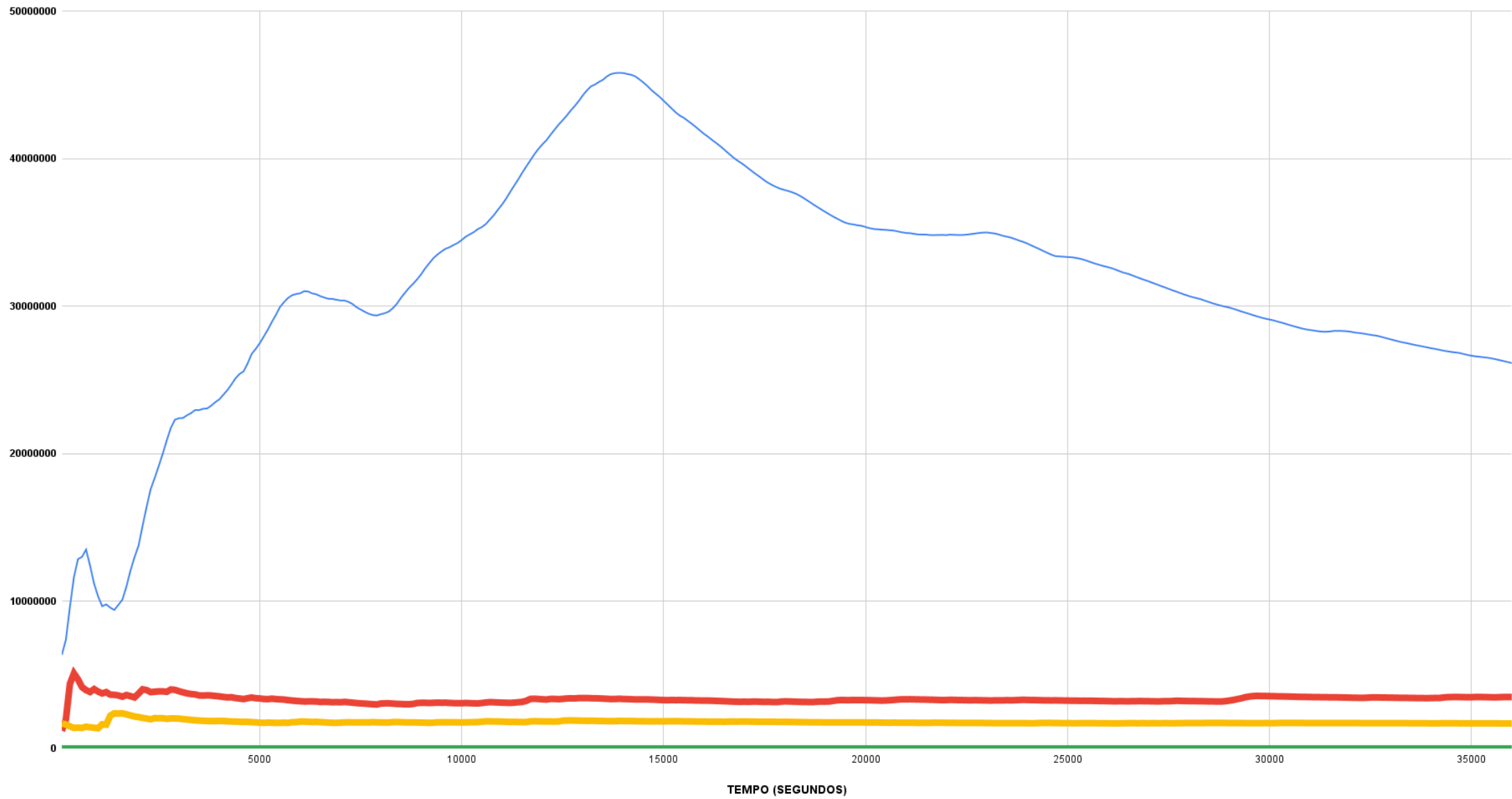
Já, para calcular o valor final de E[W] considerando os quatros cenários, assim como descrito no tópico acima, devemos pegar essa diferença e então dividir pelo tempo médio de chegadas do número de eventos.

```
double e_w_final =  
(e_w_chegada.soma_areas - e_w_saida.soma_areas) / e_w_chegada.no_eventos;
```

Cálculo E[W] final no Simulador em linguagem C

E[W] FINAL

E[W] (99%) E[W] (95%) E[W] (90%) E[W] (80%)



4. Ocupação

Assim como as medidas de little $E[N]$ e $E[W]$, a ocupação também pode ser considerada uma medida de desempenho. Mais conhecida como taxa de utilização, ela mede a fração de tempo que o sistema permanece ocupado em relação ao tempo total.

Pode-se obter matematicamente a Ocupação, calculando a chegada sobre a capacidade do sistema (λ / μ), onde:

- λ = taxa de chegada
- μ = taxa de saída

Deve-se ressaltar que o valor da taxa de utilização nunca pode exceder 1, visto que se isso ocorrer, teremos um crescimento infinito de fila.

```
double lambda = e_w_chegada.no_eventos / tempo_decorrido;
```

Cálculo λ no Simulador de linguagem C

Mesmo não definido no código-fonte modificado, podemos dizer que o valor da ocupação seria algo equivalente há:

```
Ocupação = soma_ocupacao / tempo_decorrido
```

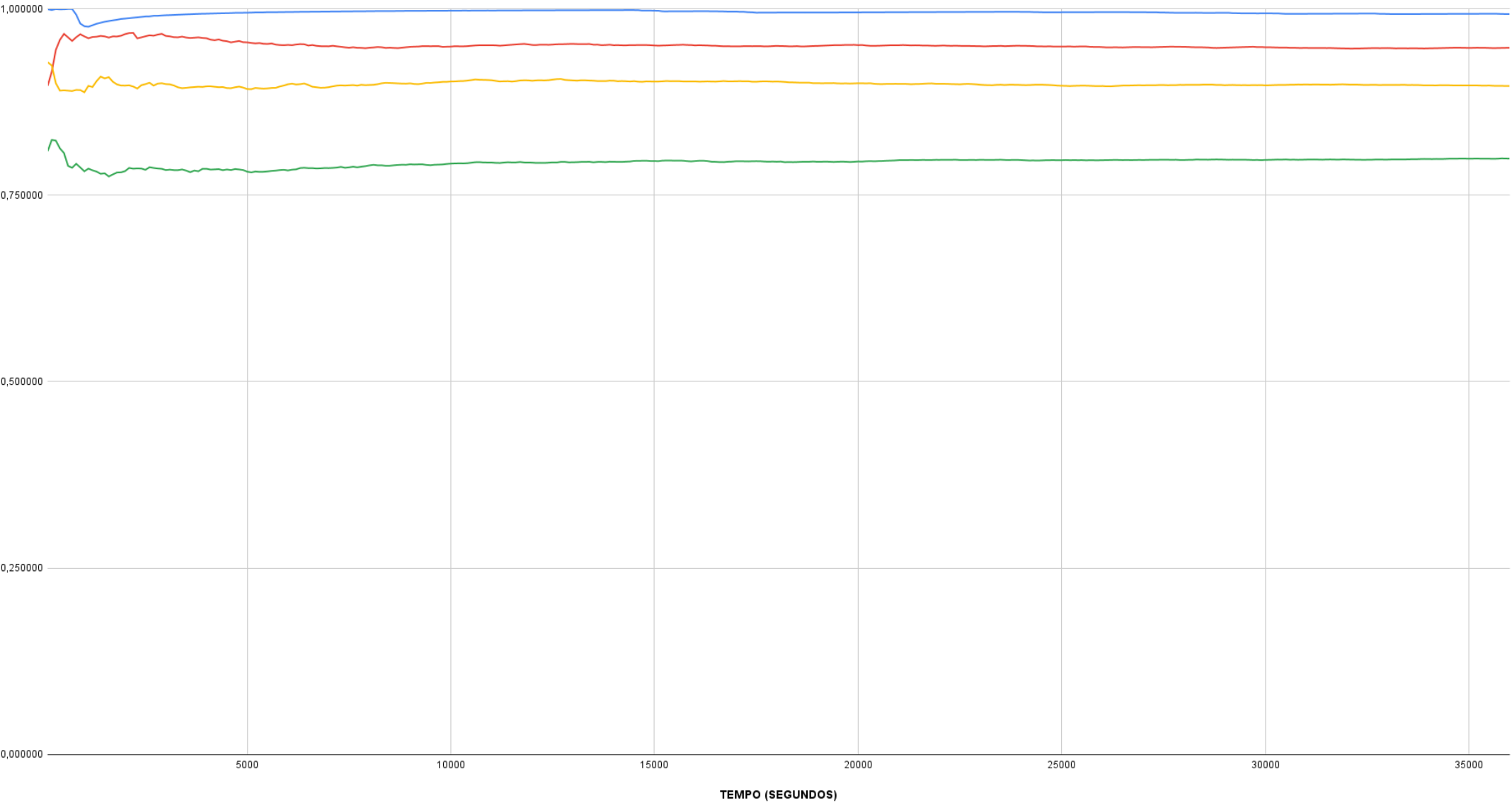
Já quanto ao cálculo da ocupação final para os quatros cenários, temos:

```
Ocupação = soma_tempo_servico / tmp_col;
```

Calculada a taxa de utilização final para todos os cenários, podemos observar que quanto maior a ocupação mais ela se aproxima de seu valor máximo, 1. Assim como podemos notar uma estabilização ao decorrer do tempo de serviço.

OCUPAÇÃO FINAL

OCUPAÇÃO (99%) OCUPAÇÃO (95%) OCUPAÇÃO (90%) OCUPAÇÃO (80%)



5. Erro de Little

Nota: para conseguir apresentar a variação do Erro de Little em um gráfico, tive que pegar os valores absolutos, convertendo-os manualmente, pois mesmo tendo criado uma função para fazer essa conversão, ao rodar o algoritmo, muitos dos valores ficaram negativos, como pode-se verificar nos arquivos txt enviados com o código-fonte do simulador.

ERRO DE LITTLE FINAL

ERRO LITTLE (99%) ERRO LITTLE (95%) ERRO LITTLE (90%) ERRO LITTLE (80%)

