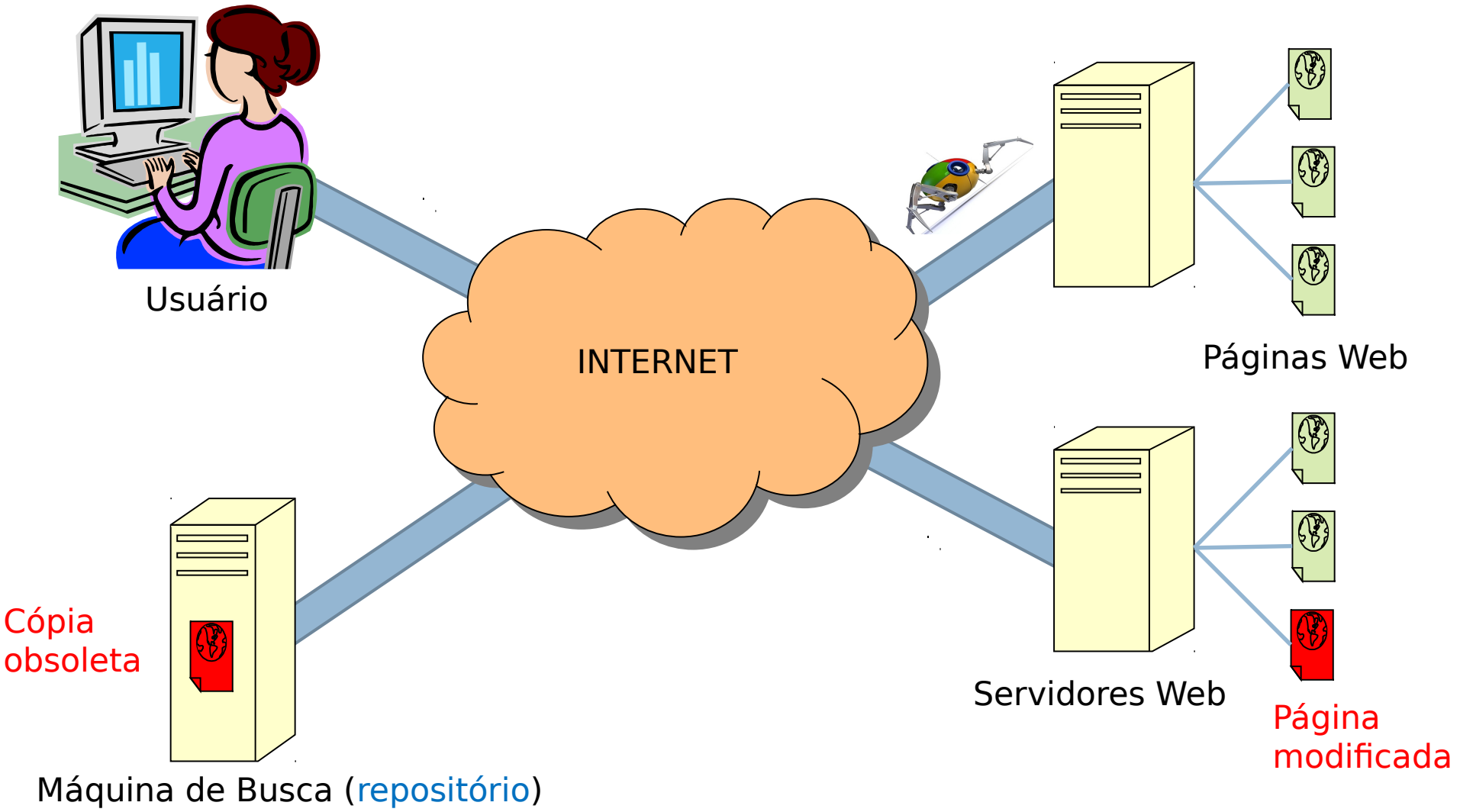
Políticas Eficientes para Revisitação de Páginas Web

Prof. Críston Souza (UFC)
Prof. Eduardo Laber (PUC-RJ)


Usuário
INTERNET
Páginas Web
Cópia obsoleta
Servidores Web
Página modificada
Máquina de Busca (repositório)

Solução: revisitar todas as páginas da Internet várias vezes por dia...

Restrições:

- Capacidade do canal de comunicação e de processamento

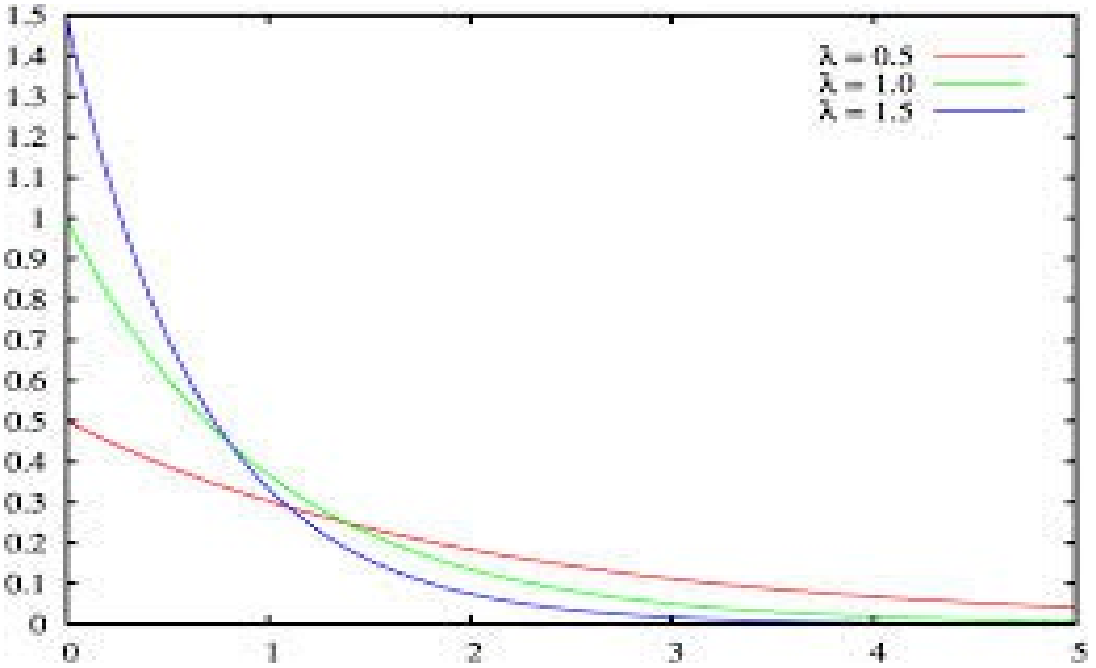
- Sobrecarga dos Servidores Web (politeness)

Objetivo:

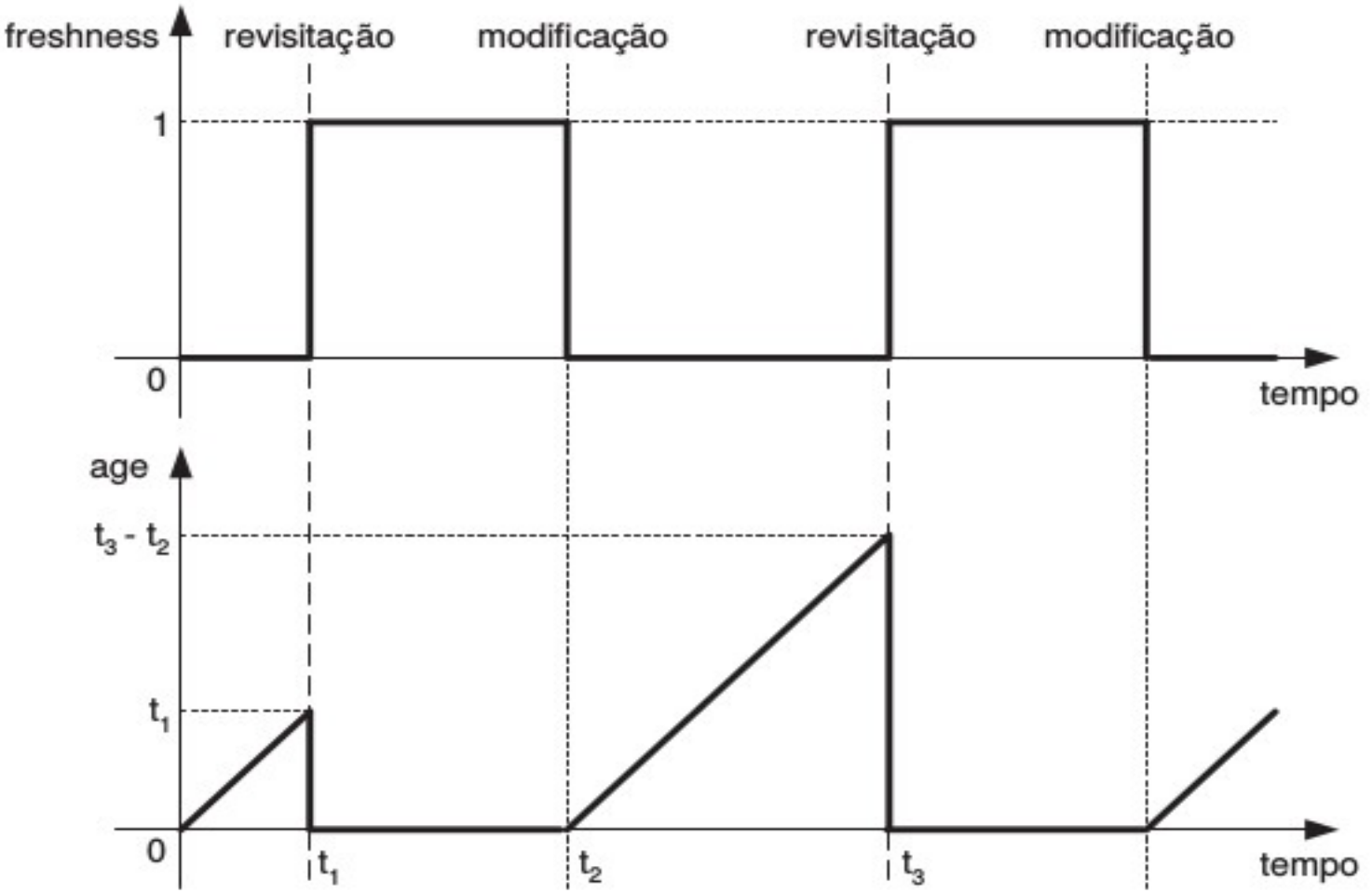
Manter o repositório o mais atualizado possível, levando em conta os limites do canal de comunicação, e respeitando um tempo mínimo entre requisições a um mesmo servidor.

Hipótese simplificadora: páginas Web são modificadas de forma independente segundo um processo de Poisson.

Muito usada na literatura e observada empiricamente.



Como medir o nível de atualização do repositório? opção pelas métricas *freshness* e *age* [Cho03].



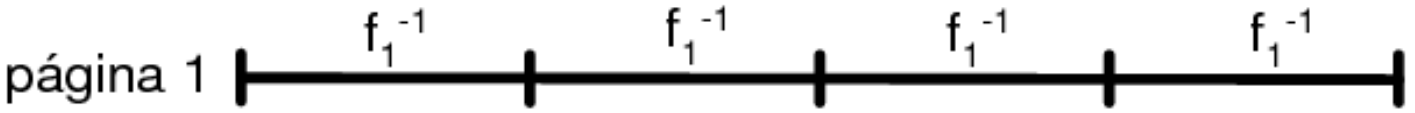
O que havia disponível na literatura?

- Ignoravam o politeness;
- Ou consumo de tempo/memória inviável.

Consumo deve ser no máximo sublinear por revisitação: mais de 10 bilhão de páginas.

Nossa abordagem: O(log(n)) por revisitação, e 0,927 aproximado.

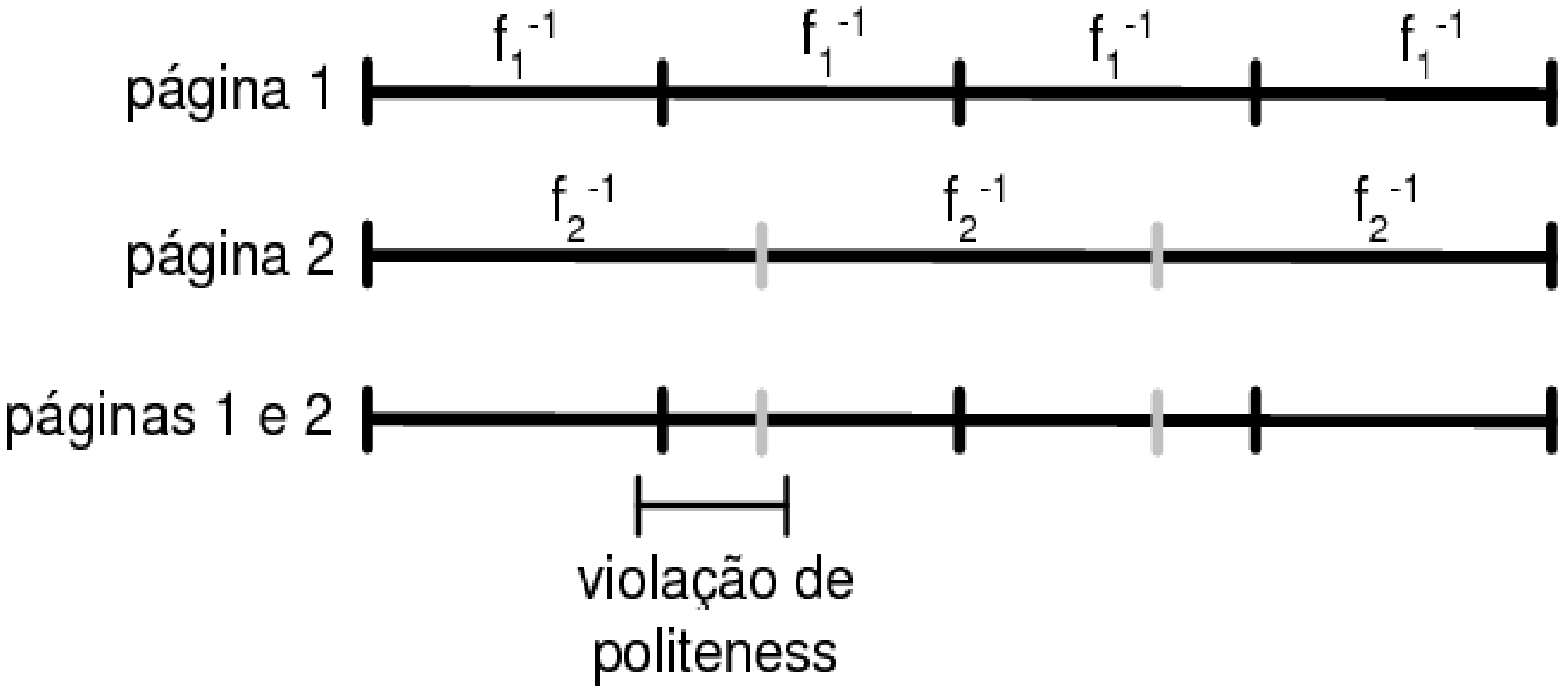
Sem a restrição de politeness, existe solução ótima para o *freshness*: política igualmente espaçada por página [Wol02,Cof98,Cho03a].



Freshness ótimo de uma página:

$$A_i^* = \frac{1 - \exp(-\lambda_i / f_i)}{\lambda_i / f_i}$$

Por que a política igualmente espaçada por página viola o politeness?



Neste caso, como decidir a frequência de revisitação de cada página? [Cho03a].

Importância da página i

Freshness ótimo da página i revisitada com frequência fi

$$maximize \quad \sum_{i \in R} w_i A_i^*(f_i)$$

$$sujeito\ à \quad \sum_{i \in R} f_i = C,$$

$$f_i \geq 0, \ para\ toda\ página\ i,$$

Capacidade total de revisitação

Frequência de revisitação da página i

Tem solução eficiente através da relaxação lagrangiana da restrição.

Como temos que respeitar o politeness, podemos obter um limite superior melhor?

Importância da página i

Freshness ótimo da página i revisitada com frequência fi

$$maximize \quad \sum_{i \in R} w_i A_i^*(f_i)$$

$$sujeito\ à \quad \sum_{i \in R} f_i = C,$$

Capacidade total de revisitação

$$\sum_{i \in R_s} f_i \leq P^{-1},\ para\ todo\ servidor\ s,$$

$$f_i \geq 0,\ para\ toda\ página\ i,$$

Frequência de revisitação da página i

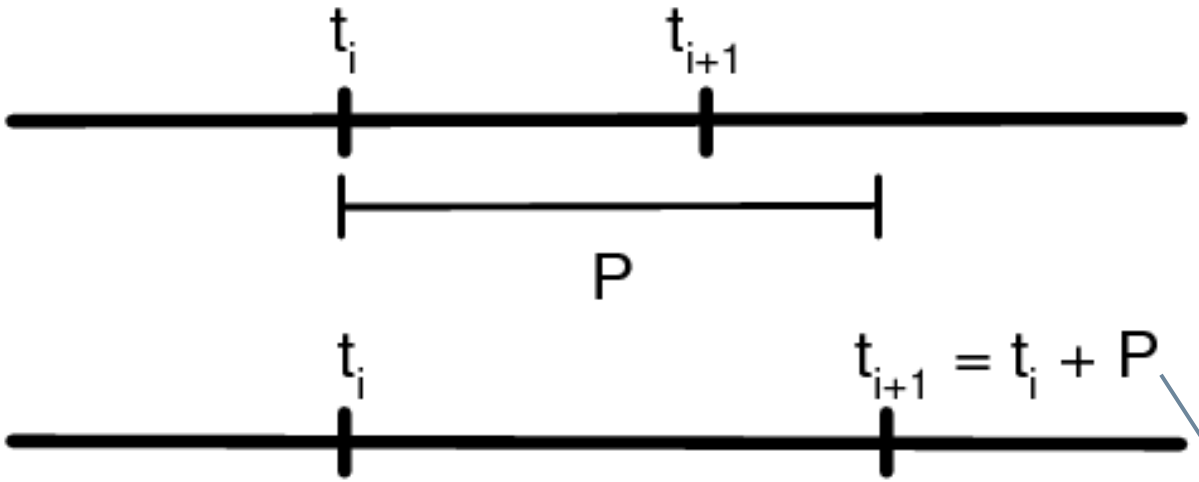
Tempo mínimo entre requisições

Também possui solução polinomial [Bre02].

E se o crawler simplesmente esperar até cumprir o tempo de politeness?

Instante da última requisição ao servidor

Instante da próxima requisição IEPP



Tempo mínimo permitido entre requisições

Repositório experimental:

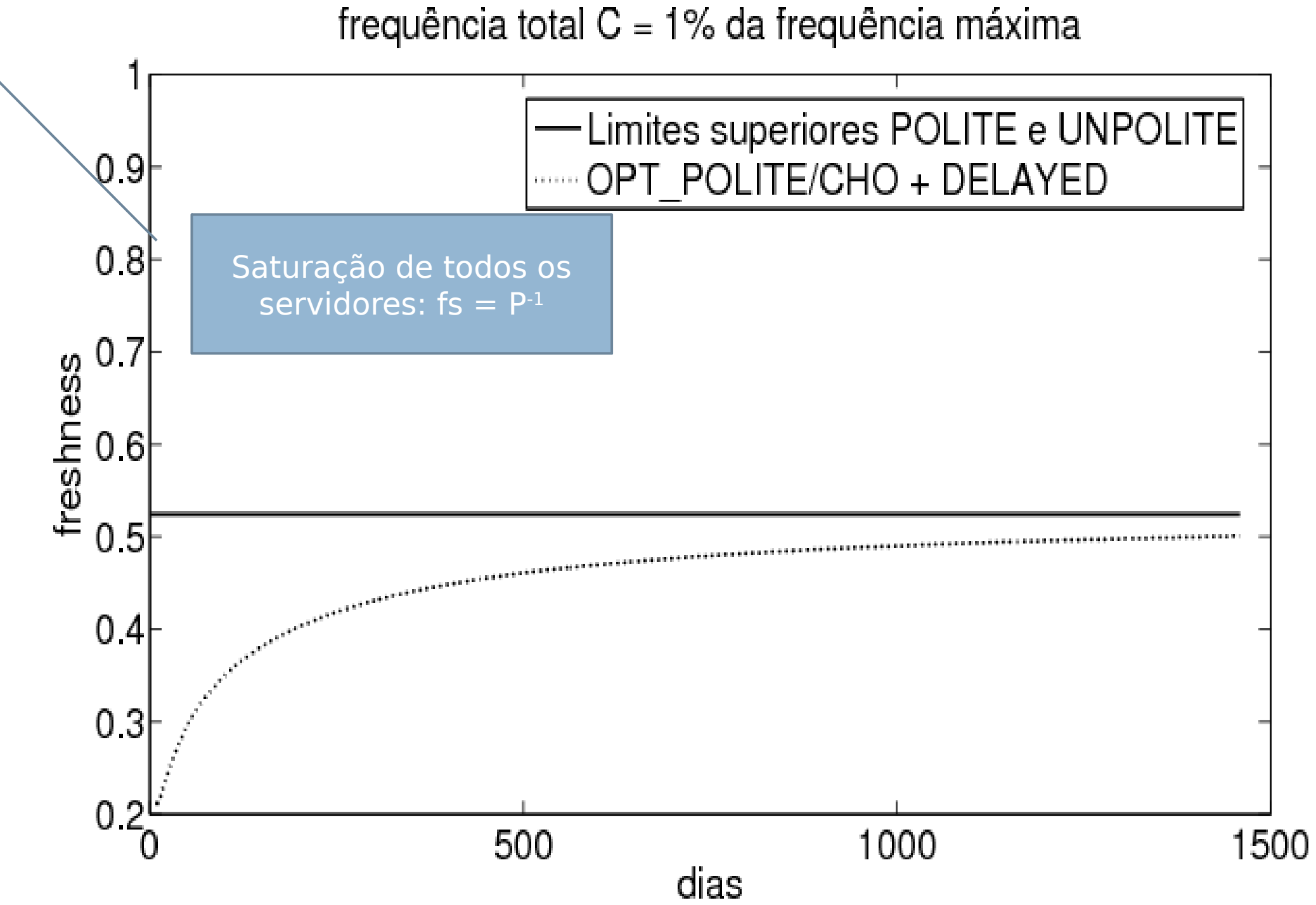
- 14,5 milhões de páginas

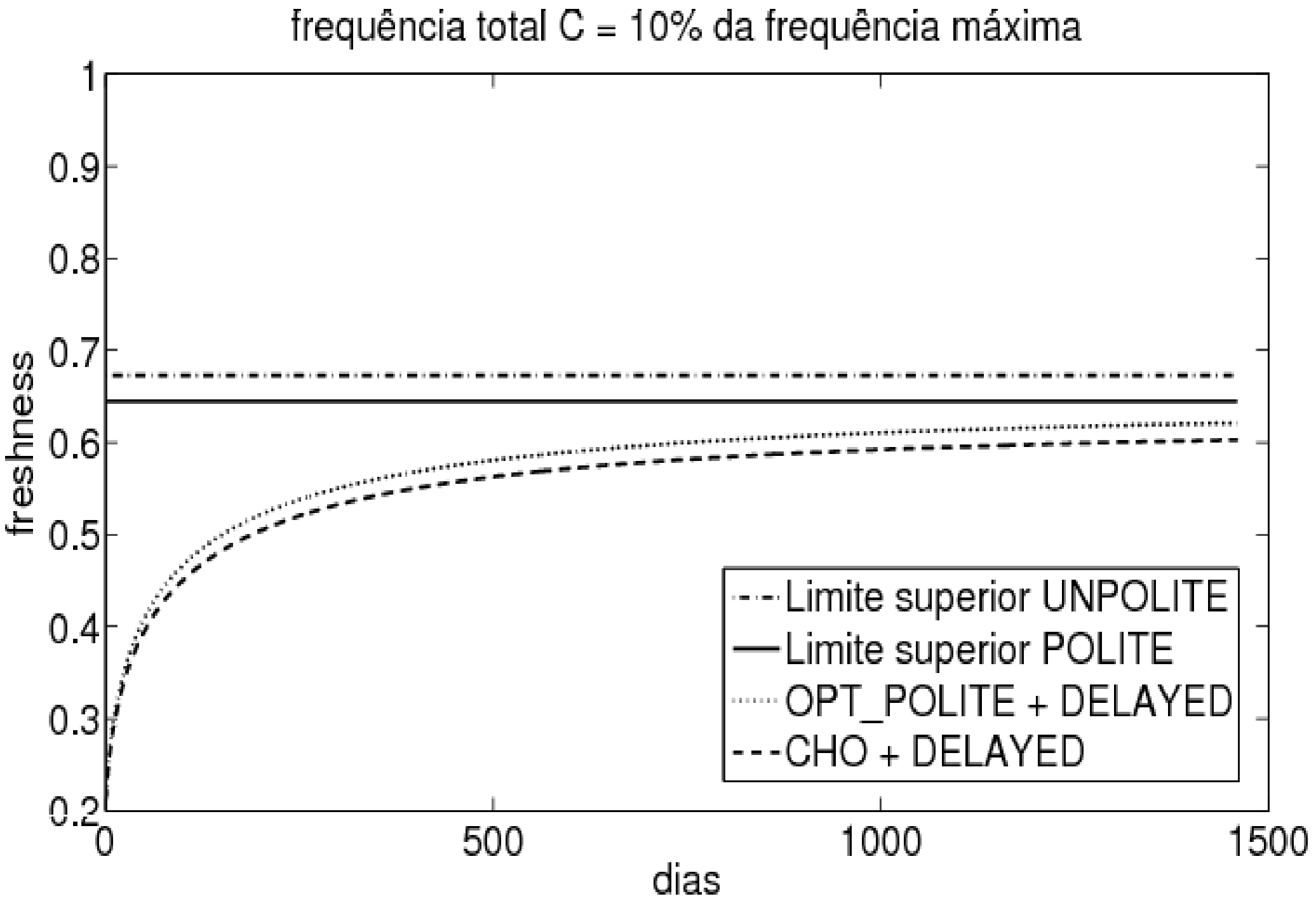
- Sementes: ~ 5.5 mil páginas do WebBase

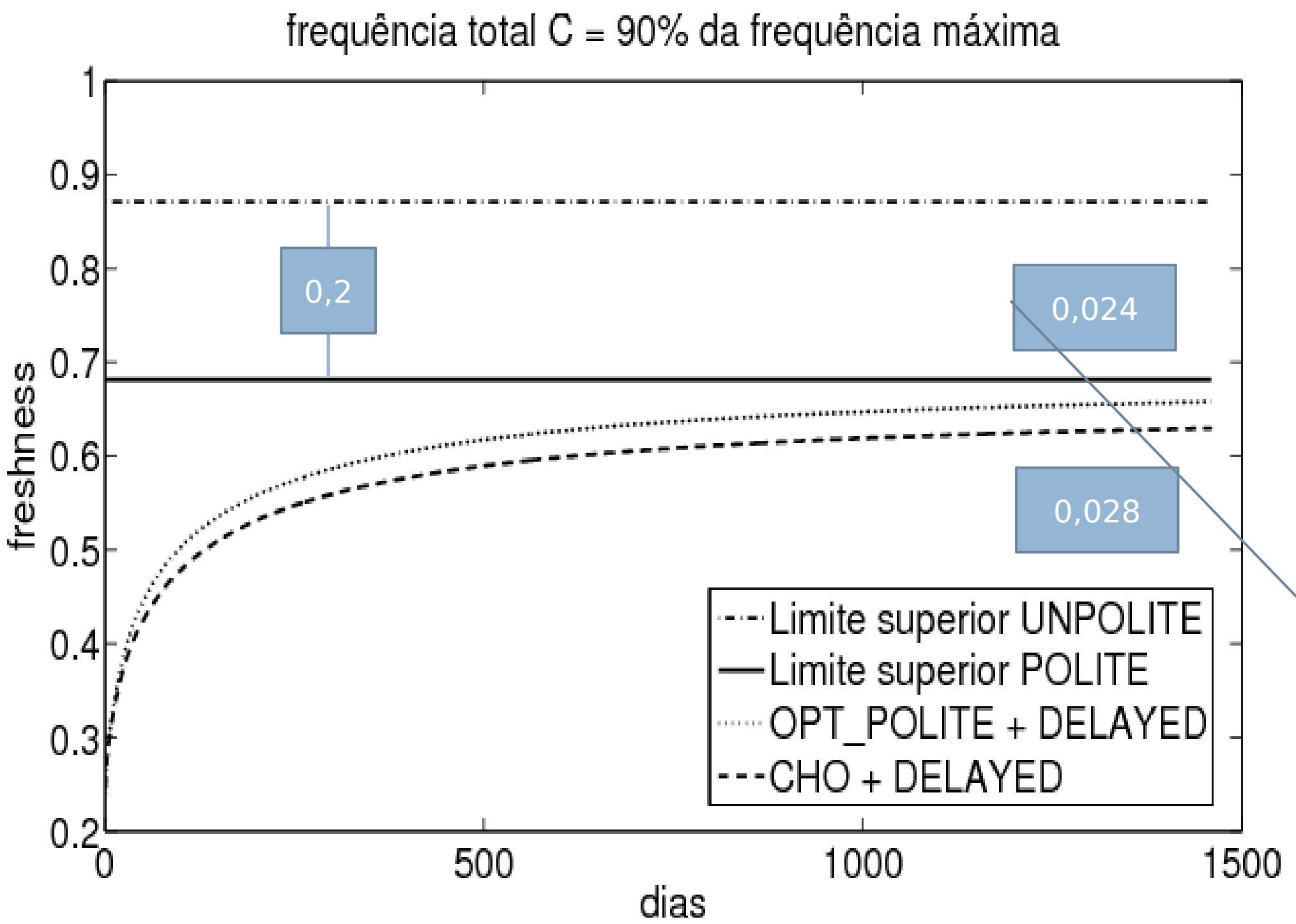
- Até 100 mil páginas por servidor

- Monitoramos ~ 360 mil páginas durante 275 dias (máximo 100 por servidor, escolhidas aleatoriamente)

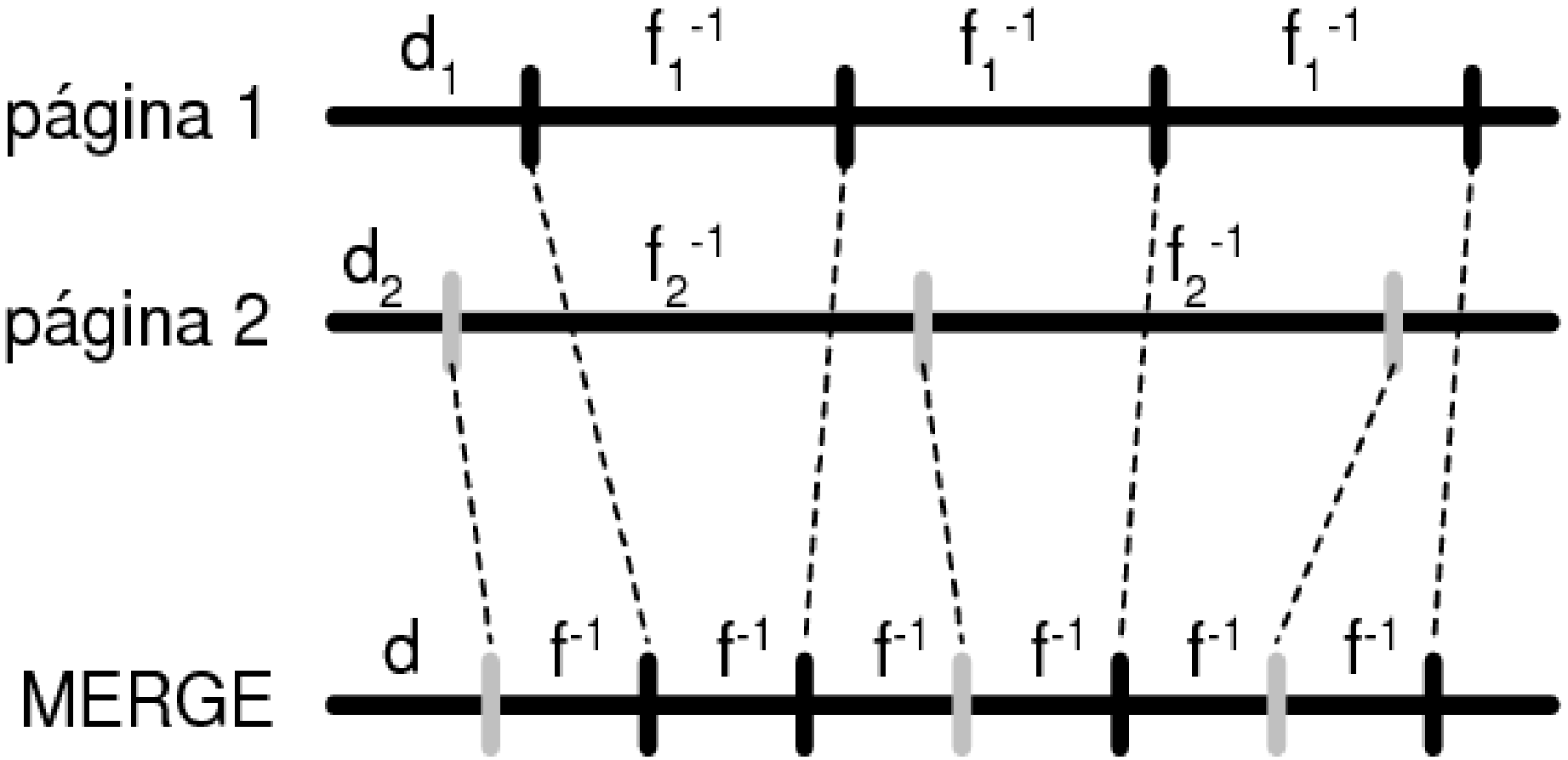
- Páginas não monitoradas: taxa sorteada dentre as monitoradas do mesmo servidor.


frequência total C = 1% da frequência máxima
Limites superiores POLITE e UNPOLITE
OPT_POLITE/CHO + DELAYED
Saturação de todos os servidores: fs = P-1
freshness
dias
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
500
1000
1500


frequência total C = 10% da frequência máxima
freshness
dias
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
500
1000
1500
Limite superior UNPOLITE
Limite superior POLITE
OPT_POLITE + DELAYED
CHO + DELAYED


frequência total C = 90% da frequência máxima
freshness
dias
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
500
1000
1500
0,2
0,024
0,028
Limite superior UNPOLITE
Limite superior POLITE
OPT_POLITE + DELAYED
CHO + DELAYED

Mantendo as requisições igualmente espaçadas por servidor, resta apenas decidir a ordem de revisitação...

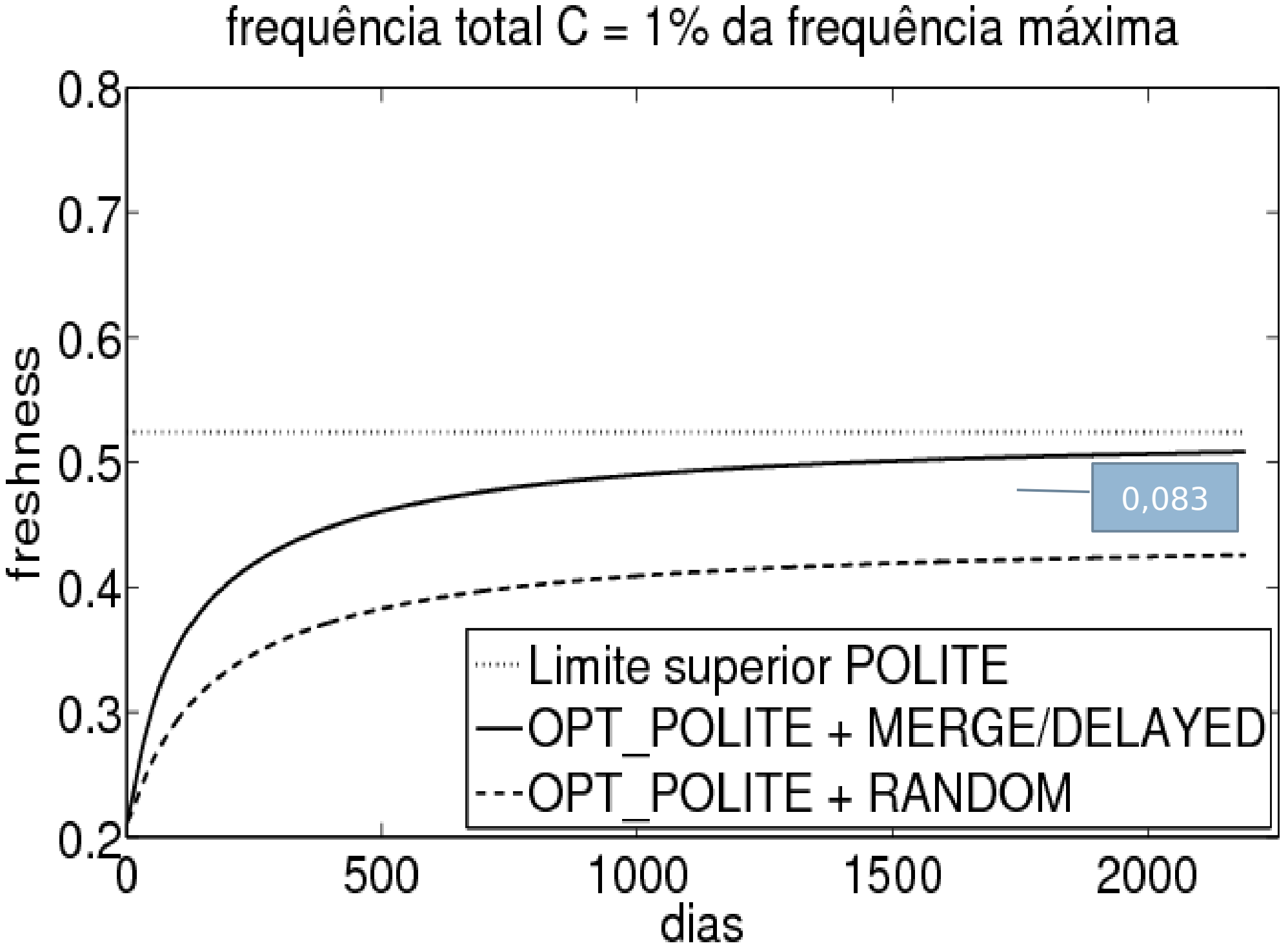


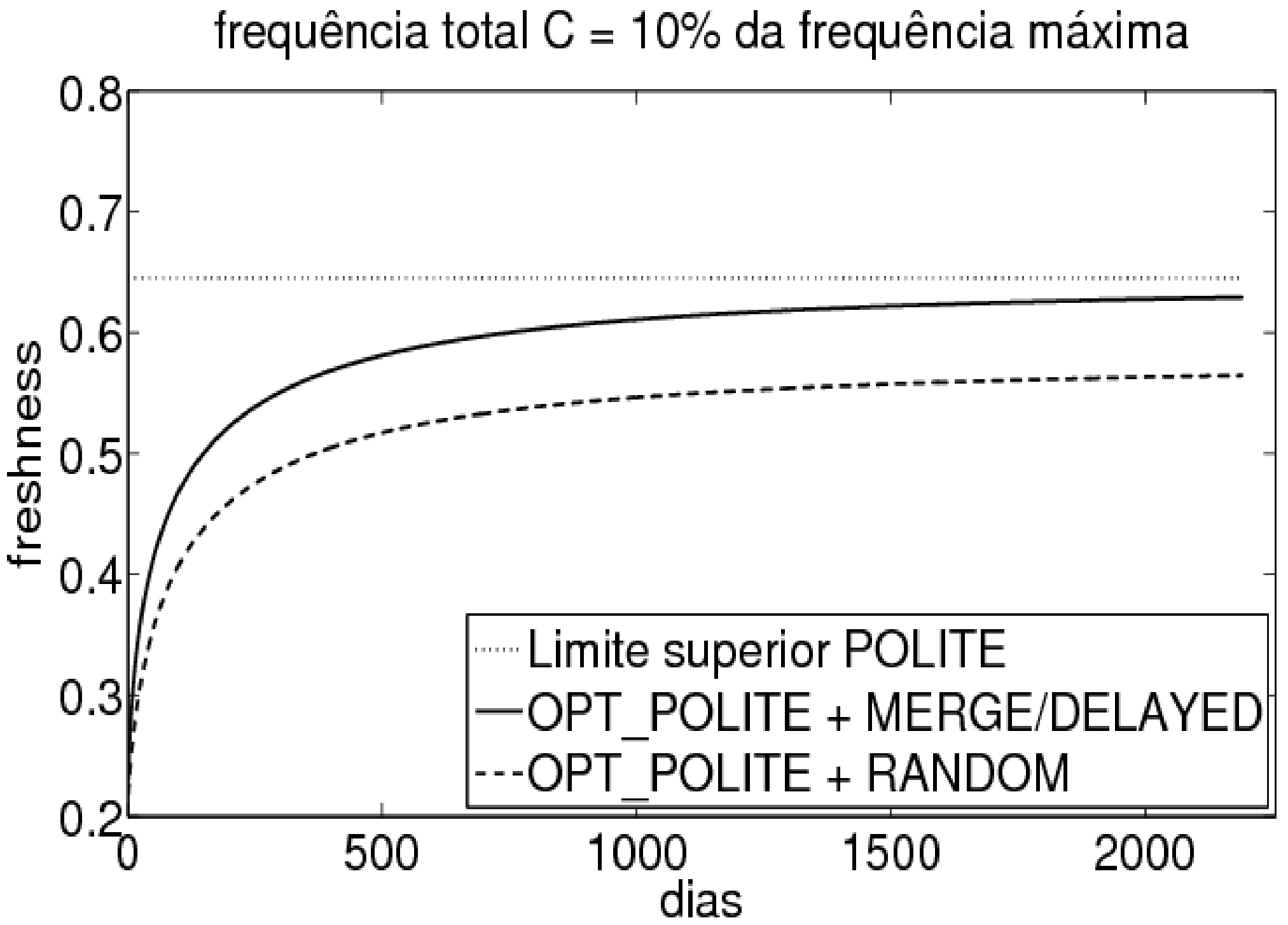
Qual a complexidade de escolha da próxima página a revisitar?

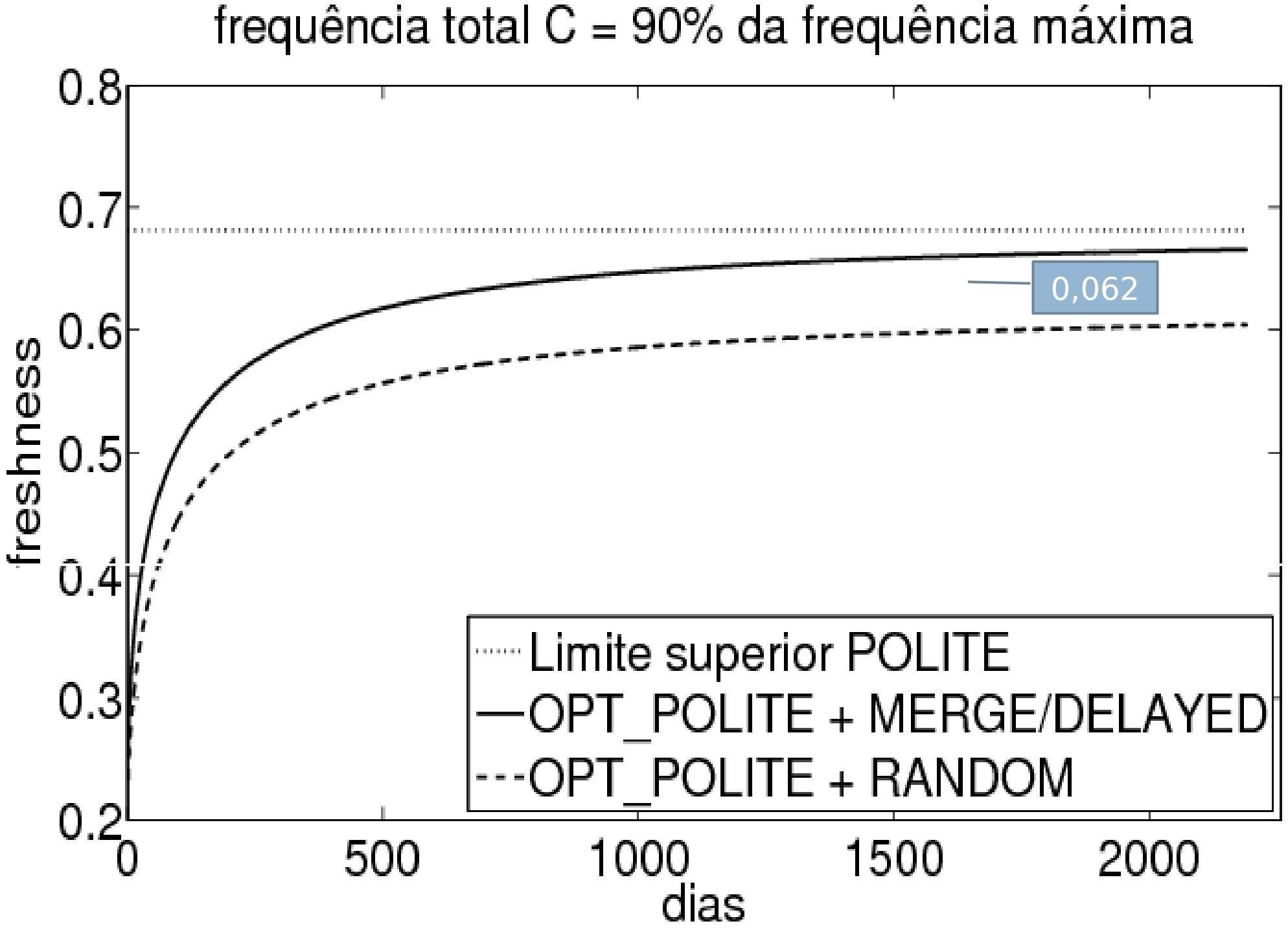
Utilizando uma heap de servidores, e uma heap de páginas para cada servidor:

Tempo: $O(\log(m) + \log(n)) = O(\log n)$
Espaço: $O(m + n) = O(n)$

m – número de servidores
n – número de páginas


frequência total C = 1% da frequência máxima
freshness
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
500
1000
1500
2000
dias
0,083
Limite superior POLITE
OPT_POLITE + MERGE/DELAYED
OPT_POLITE + RANDOM

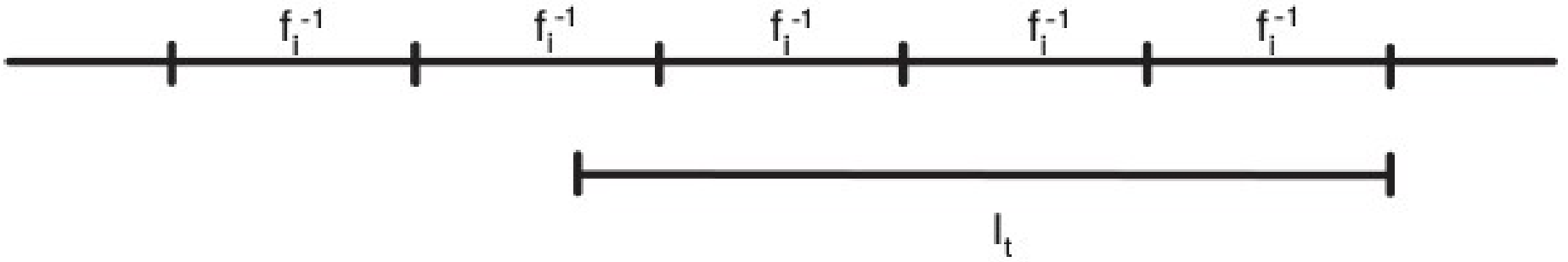

frequência total C = 10% da frequência máxima
freshness
dias
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
500
1000
1500
2000
Limite superior POLITE
OPT_POLITE + MERGE/DELAYED
OPT_POLITE + RANDOM


frequência total C = 90% da frequência máxima
freshness
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
500
1000
1500
2000
dias
0,062
Limite superior POLITE
OPT_POLITE + MERGE/DELAYED
OPT_POLITE + RANDOM

Qual o *freshness* de uma página revisitada de acordo com a política MERGE?

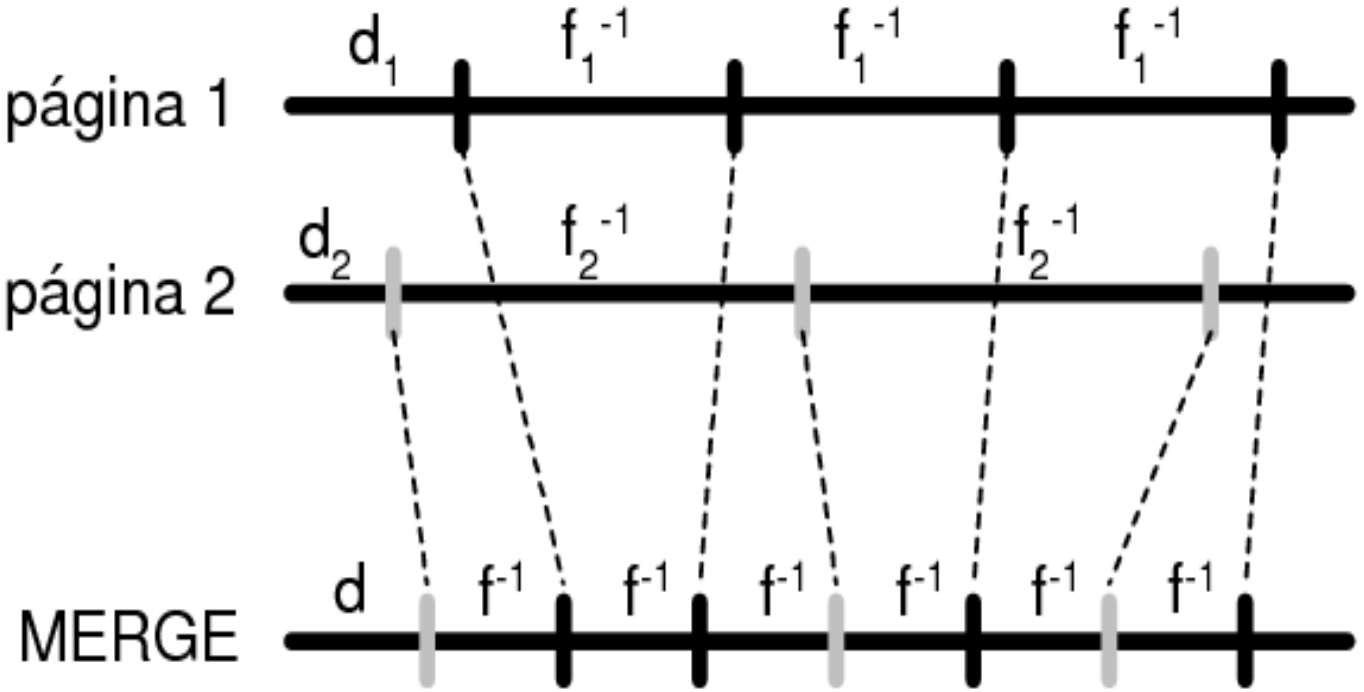
Quantas revisitações da página i ocorrem em um intervalo de tamanho t?

$X_i(t) = piso(tf_i) + Y_i(t)$, onde $Y_i(t)$ é uma V.A. de Bernoulli com prob. $tf_i - piso(tf_i)$



Qual o *freshness* de uma página revisitada de acordo com a política MERGE?

O tempo entre revisitações de uma página depende do número de outras páginas selecionadas neste intervalo (soma de V.A. de Bernoulli).



Qual o *freshness* de uma página revisitada de acordo com a política MERGE?

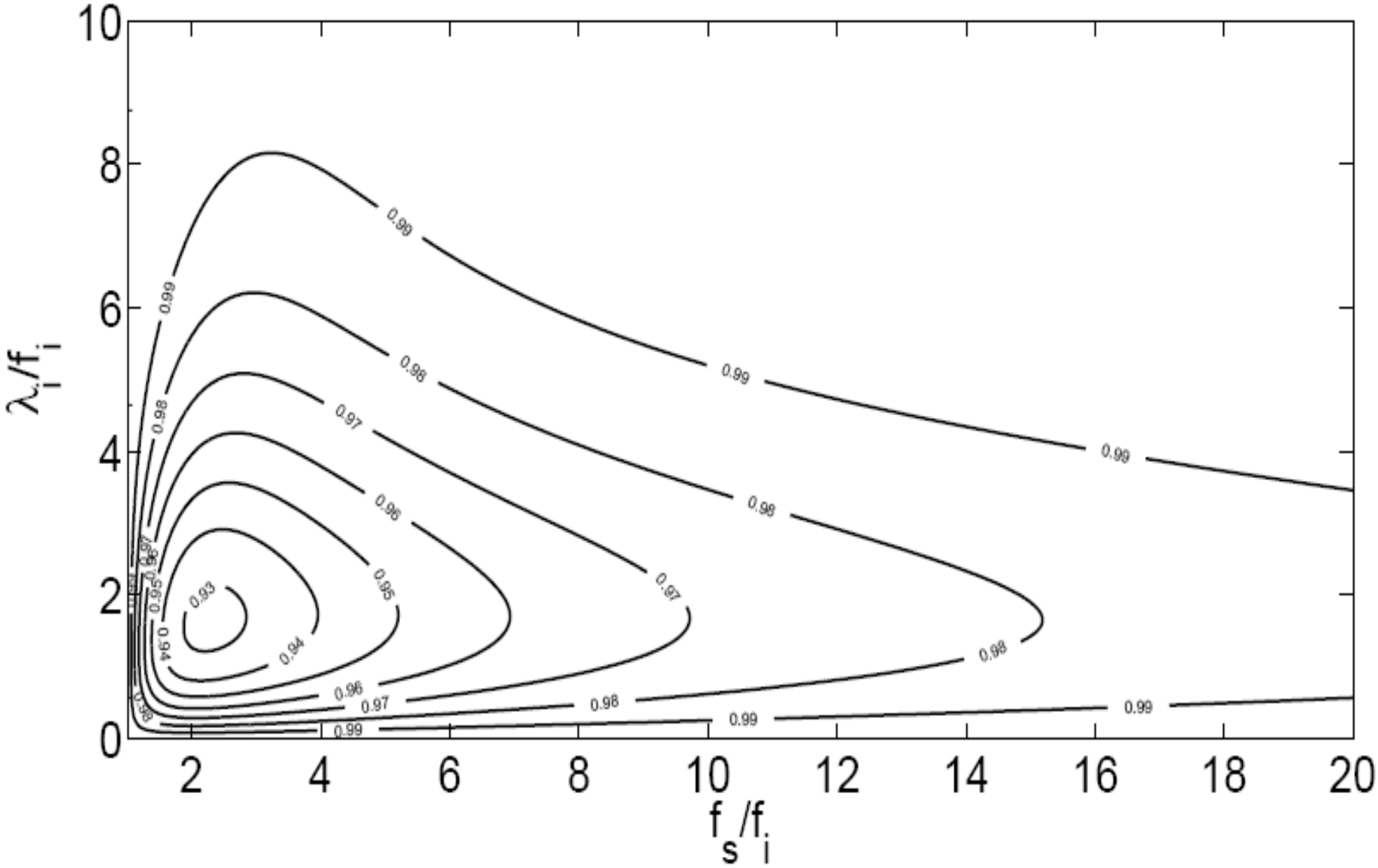
Frequência de revisitação da página i

Taxa de modificação da página i

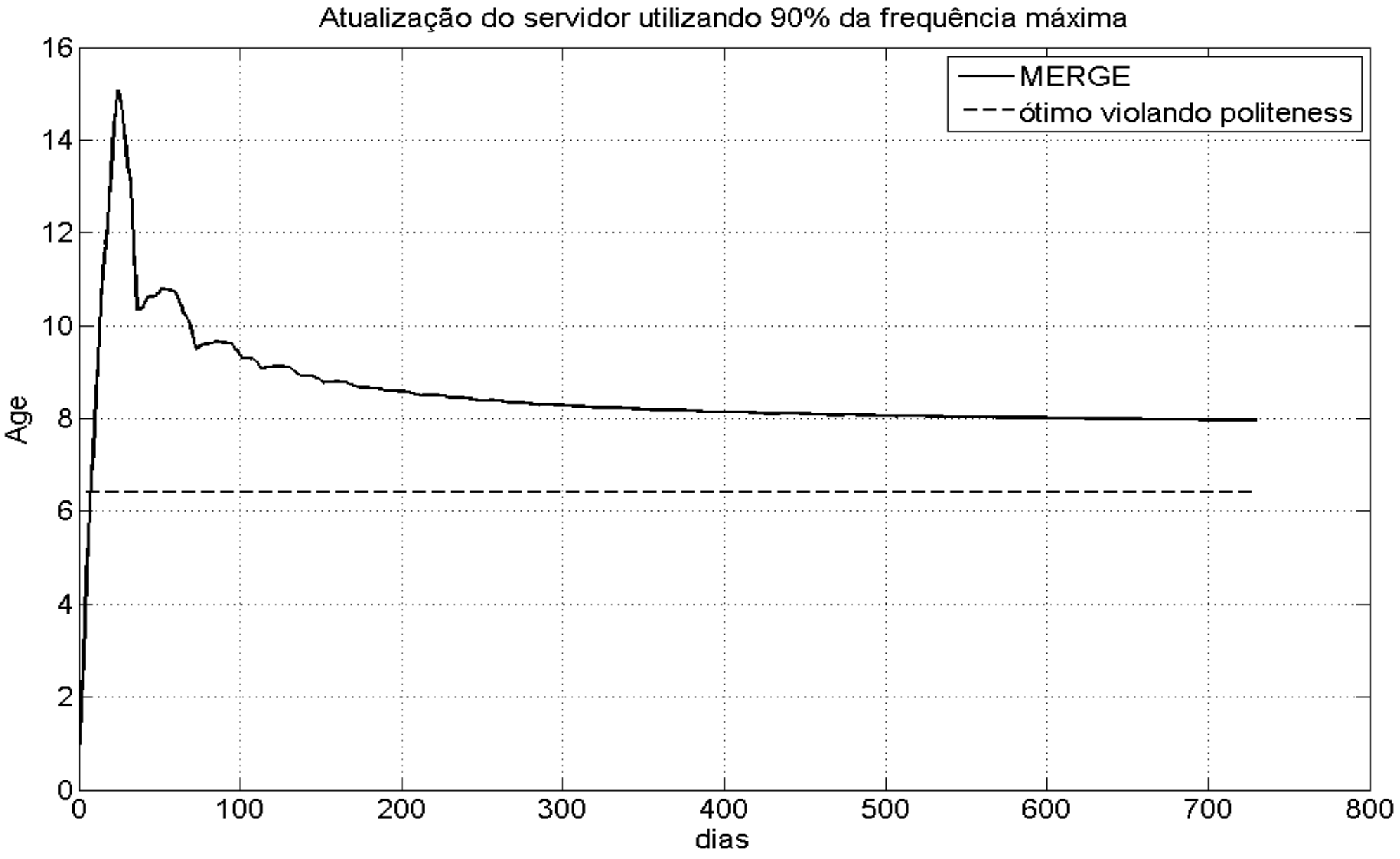
$$A_i \geq \frac{f_i}{\lambda_i}\left(1-\exp\left[-\frac{\lambda_i}{f_s}+\left(\exp\left(-\frac{\lambda_i}{f_s}\right)-1\right)\left(\frac{f_s-f_i}{f_i}\right)\right]\right)$$

Frequência total de revisitação às páginas so servidor que hospeda i

Fator de aproximação Ai / A*i > 0,927



Podemos obter um fator de aproximação tão bom para a métrica *age*? Não. Dif > 20%.



Como paralelizar a política MERGE?

Podemos ter as heaps de páginas em máquinas distintas [WIRE].

criston@ufc.br