# Advanced topics on Algorithms

Luciano Gualà
www.mat.uniroma2.it/~guala/

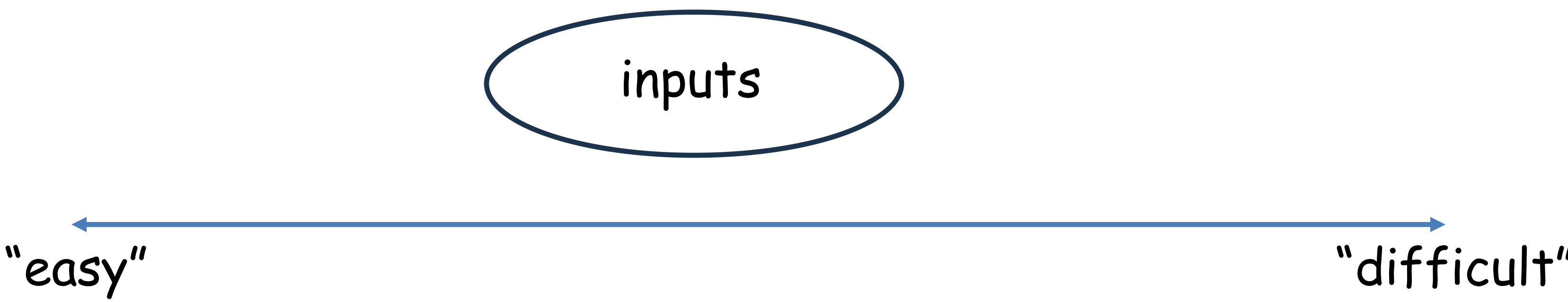
# Beyond Worst-Case Analysis
# Episode II

# parameterized analysis of online paging

# Parameterized analysis

Two-step approach:

1. impose a natural parameterization of all inputs
2. analyze the performance of an algorithms as a function of the parameter




goal: more fine-grained analysis than traditional worst-case analysis

Ideal: the parameter is independent of the description of the algorithm, and is referenced only in the algorithm's analysis

Why bother?

to extract advice about when (on which types of inputs) you should use the algorithm

# Working Set Model [Denning '68]

Let $f: N \to N$.

Sequence z conforms to f if $\forall$ w, $\forall$ window (subsequence of z) of size w the number of distinct page requests is at most f(w).

Example: if f is the identity function, $f(n)=n\ \forall n$.

⇒ no restriction

Example: if $f(2)=1$

⇒ only the "constant" sequence

Example:

$f(n) = \sqrt{n}$ or $f(n) = \log n$

idea #1: parameterized performance by $f$

Issue: it can be proved that for any non-trivial $f$ the competitive ratio is $k$

idea #2: to evaluate performance use the page fault rate (#of faults)/$|z|$

Theorem [Albers, Favrholdt, Giel, STOC 02]

(a) For every concave function $f$, for every cache size $k$, and for every deterministic online algorithm $A$, the worst-case page fault rate (over sequences that conform to $f$) is $\geq \alpha_f(k)$.
(b) For every concave function $f$, and for every cache size $k$, the worst-case page fault rate (over sequences that conform to $f$) of the LRU algorithm is $\leq \alpha_f(k)$.
(c) There exists a concave function $f$, a cache size $k$, and a request sequence $z$ that conforms to $f$ such that the page fault rate of the FIFO algorithm on $z$ is $> \alpha_f(k)$.

never a "jump": $f(n+1) \le f(n)+1$

| f(n) | 1 | 2 | 3 | 3 | 4 | 4 | 4 | 5 | ... |
|---|---|---|---|---|---|---|---|---|---|
| n | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... |

$m_j$: number of (consecutive) values of $n$ for which $f(n) = j$

concave function: $m_1 \le m_2 \le m_3 \le m_4 \le \ldots$

$f^{-1}(m)$: smallest value of $n$ for which $f(n) = m$

$$\alpha_f(k) = \frac{k-1}{f^{-1}(k+1)-2}$$

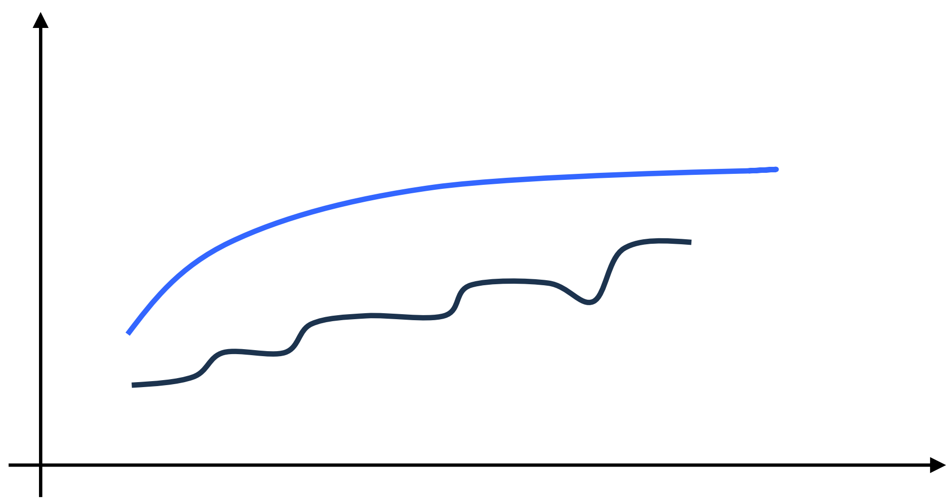
obs: if in a window there are $p$ distinct pages, then the size of the window is at least $f^{-1}(p)$.

$$\alpha_f(k) = \frac{k-1}{f^{-1}(k+1)-2}$$

Example: if f is the identity function, $f(n)=n\ \forall n$. ⇒ $\alpha_f(k) = 1$

Example: if $f(n) \approx \sqrt{n}$ ⇒ $\alpha_f(k) \approx 1/k$

Example: if $f(n) \approx \log n$ ⇒ $\alpha_f(k) \approx k/2^k$



obs: when the empirically observed working set function f' is not concave, one can choose any concave function f with $f'(n) \leq f(n)$.

## proof of (a)

fix k, f and A, and assume #of existing pages N=k+1

sequence z consists of an arbitrarily large number t of phases

a phase consists of k-1 blocks:

$m_2$ requests of page $p_2$; ($p_2$: page missing from A's cache at the start of block 1)

$m_3$ requests of page $p_3$; ($p_3$: page missing from A's cache at the start of block 2)

.
.
.

$m_k$ requests of page $p_k$; ($p_k$: page missing from A's cache at the start of block k-1)

#of faults per phase: k-1

length of a phase: $m_2 + m_3 + \ldots + m_k = f^{-1}(k+1)-2$

Claim: z conforms to f

(Exercise: check it)

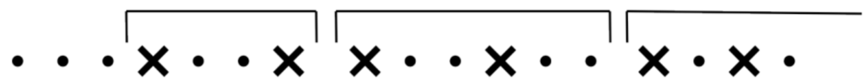
obs: $m_1 + m_2 + \ldots + m_k = f^{-1}(k+1)-1$

| f(n) | 1 | 2 | 3 | 3 | 4 | 4 | 4 | 5 | ... |
|---|---|---|---|---|---|---|---|---|---|
| n | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... |

## proof of (b)

fix k and f, and consider a sequence z that conforms to f

divide z into block of k-1 faults (a block begins with a fault and ends with the request just before the next block)



example for k=3

**Claim:** Consider a block other than the first or last. Consider the page requests in this block, together with the requests immediately before and after this block. These requests are for at least k + 1 distinct pages.

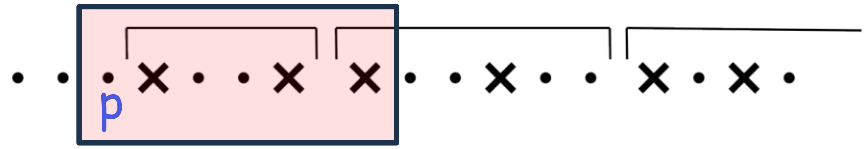
⇒ every block contains at least $f^{-1}(k+1)-2$ requests

⇒ page fault rate is at most $\alpha_f(k)$ (ignoring the vanishing additive error due to the first and last blocks)

## proof of (b)

fix k and f, and consider a sequence z that conforms to f

divide z into block of k-1 faults (a block begins with a fault and ends with the request just before the next block)




example for k=3

Claim: Consider a block other than the first or last. Consider the page requests in this block, together with the requests immediately before and after this block. These requests are for at least k + 1 distinct pages.

## proof of Claim

fix a block, and let p be the page requested immediately prior to this block

case 1: k faults occur on distinct pages that are all different from p

⇒ k+1 distinct pages (p + k faults)

case 2: there are 2 faults occur on same page $q \neq p$

⇒ LRU implies k+1 distinct pages (q and the k other distinct requests between the two faults on q)

case 3: one of these faults is on p

⇒ LRU implies k+1 distinct pages (p and the k other distinct requests between the first request of p and the fault on p)

■

## proof of (c)

$k=4$ set of pages = {0,1,2,3,4}

| f(n) | 1 | 2 | 3 | 3 | 4 | 4 | 5 | 5 | ... |
|---|---|---|---|---|---|---|---|---|---|
| n | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... |

$\alpha_f(k)= 3/5= 60\%$

proof of (c)

k=4 set of pages = {0,1,2,3,4}

| f(n) | 1 | 2 | 3 | 3 | 4 | 4 | 5 | 5 | ... |
|---|---|---|---|---|---|---|---|---|---|
| n | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... |

$\alpha_f(k)$= 3/5= 60%

sequence z = 1 0 2 0 3 0 4 0 1 0 2 0 3 0 4 0 ....

x x x v x v x

cache

| |
|---|
| 1 |
| 0 |
| 2 |
| 3 |

# proof of (c)

k=4 set of pages = {0,1,2,3,4}

| f(n) | 1 | 2 | 3 | 3 | 4 | 4 | 5 | 5 | ... |
|---|---|---|---|---|---|---|---|---|---|
| n | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... |

$\alpha_f(k) = 3/5 = 60\%$

sequence z = 1 0 2 0 3 0 4 0 1 0 2 0 3 0 4 0 ....

× × × v × v × v ×

cache

| |
|---|
| 4 |
| 0 |
| 2 |
| 3 |

## proof of (c)

k=4 set of pages = {0,1,2,3,4}

| f(n) | 1 | 2 | 3 | 3 | 4 | 4 | 5 | 5 | ... |
|---|---|---|---|---|---|---|---|---|---|
| n | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... |

$\alpha_f(k)$= 3/5= 60%

sequence z = 1 0 2 0 3 0 4 0 1 0 2 0 3 0 4 0 ....

× × × v × v × v × ×

cache

| |
|---|
| 4 |
| 1 |
| 2 |
| 3 |

## proof of (c)

k=4    set of pages = {0,1,2,3,4}

| f(n) | 1 | 2 | 3 | 3 | 4 | 4 | 5 | 5 | ... |
|---|---|---|---|---|---|---|---|---|---|
| n | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... |

$\alpha_f(\mathbf{k})$= 3/5= 60%

sequence z = 1 0 2 0 3 0 4 0 1 0 2 0 3 0 4 0 ....

× × × v × v × v × × ×

cache

| |
|---|
| 4 |
| 1 |
| 0 |
| 3 |

# proof of (c)

k=4 set of pages = {0,1,2,3,4}

| f(n) | 1 | 2 | 3 | 3 | 4 | 4 | 5 | 5 | ... |
|---|---|---|---|---|---|---|---|---|---|
| n | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... |

$\alpha_f(k)$= 3/5= 60%

sequence z = 1 0 2 0 3 0 4 0 1 0 2 0 3 0 4 0 ....

× × × v × v × v × × × v ×

cache

| |
|---|
| 4 |
| 1 |
| 0 |
| 2 |

# proof of (c)

k=4 set of pages = {0,1,2,3,4}

| f(n) | 1 | 2 | 3 | 3 | 4 | 4 | 5 | 5 | ... |
|---|---|---|---|---|---|---|---|---|---|
| n | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... |

$\alpha_f(\mathbf{k})$= 3/5= 60%

sequence z = 1 0 2 0 3 0 4 0 1 0 2 0 3 0 4 0 ....

× × × v × v × v × × × v × v ×

cache

| |
|---|
| 3 |
| 1 |
| 0 |
| 2 |

## proof of (c)

k=4 set of pages = {0,1,2,3,4}

| f(n) | 1 | 2 | 3 | 3 | 4 | 4 | 5 | 5 | ... |
|---|---|---|---|---|---|---|---|---|---|
| n | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... |

$\alpha_f(k)$= 3/5= 60%

sequence z = 1 0 2 0 3 0 4 0 1 0 2 0 3 0 4 0 ....

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 2 | 0 | 3 | 0 | 4 | 0 | 1 | 0 | 2 | 0 | 3 | 0 | 4 | 0 |
| x | x | x | v | x | v | x | v | x | x | x | v | x | v | x | v |

cache

| |
|---|
| 3 |
| 4 |
| 0 |
| 2 |

FIFO fault rate= 5/8 = 62,5%