Path Compression and Making the Inverse Ackermann Function Appear Natural(ly)

Raimund Seidel

Universität des Saarlandes

Bob Tarjan 1975

Theorem:

Any sequence of m Union, Find operations in a universe of n elements that uses linking by rank and path compression takes time at most

$$O(\ m \cdot \alpha(m,n) + n\ )$$

Bob Tarjan 1975

Theorem:

Any sequence of m Union, Find operations in a universe of n elements that uses linking by rank and path compression takes time at most

$$O(\ m\cdot\alpha(m,n)\ +n\ )$$

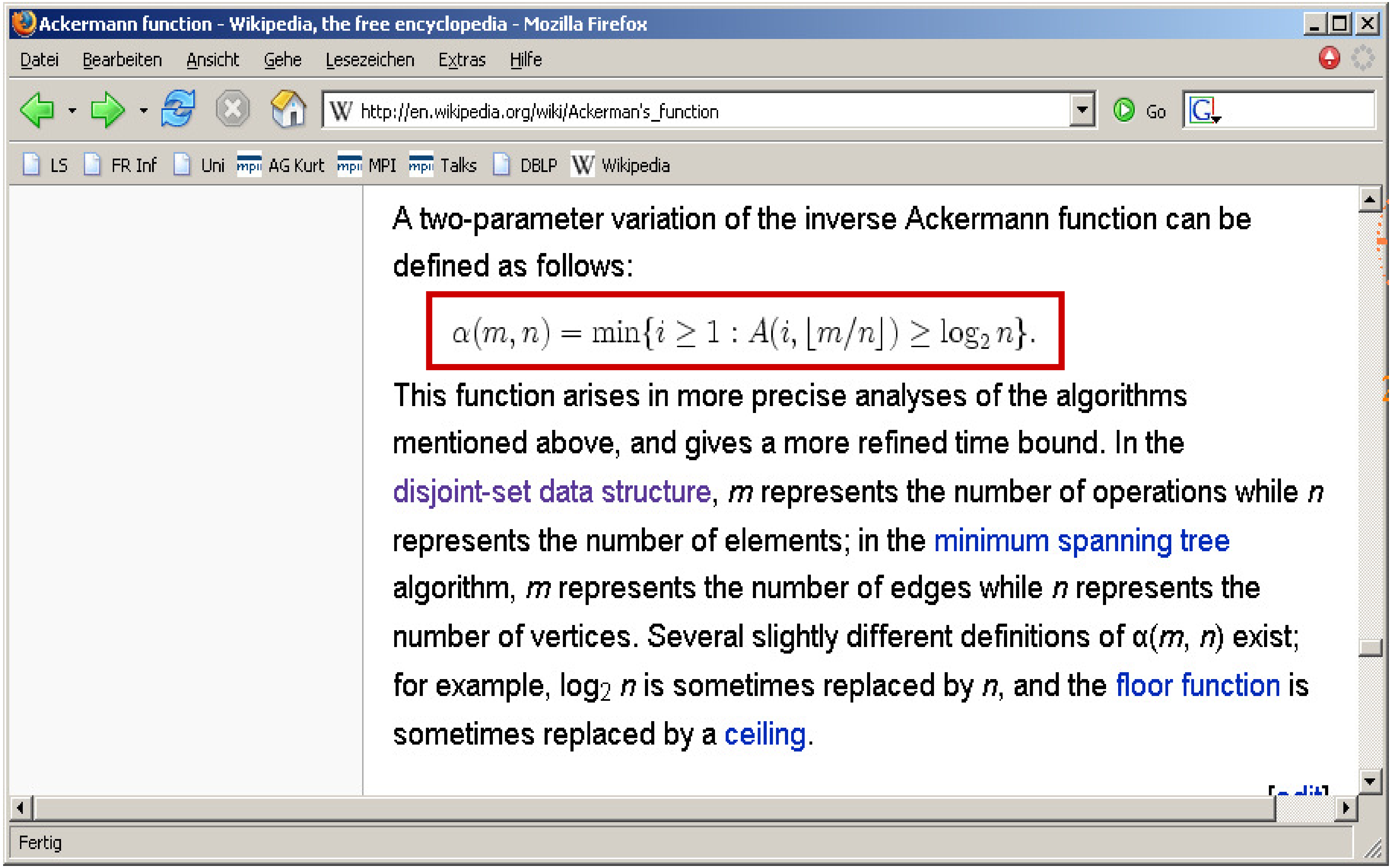
where $\alpha(m,n)$ is the "Functional Inverse" of the Ackermann Function.

What is this $\alpha(m,n)$??

What is this $\alpha(m,n)$??

Why does this $\alpha(m,n)$ appear in the analysis of path compression ??

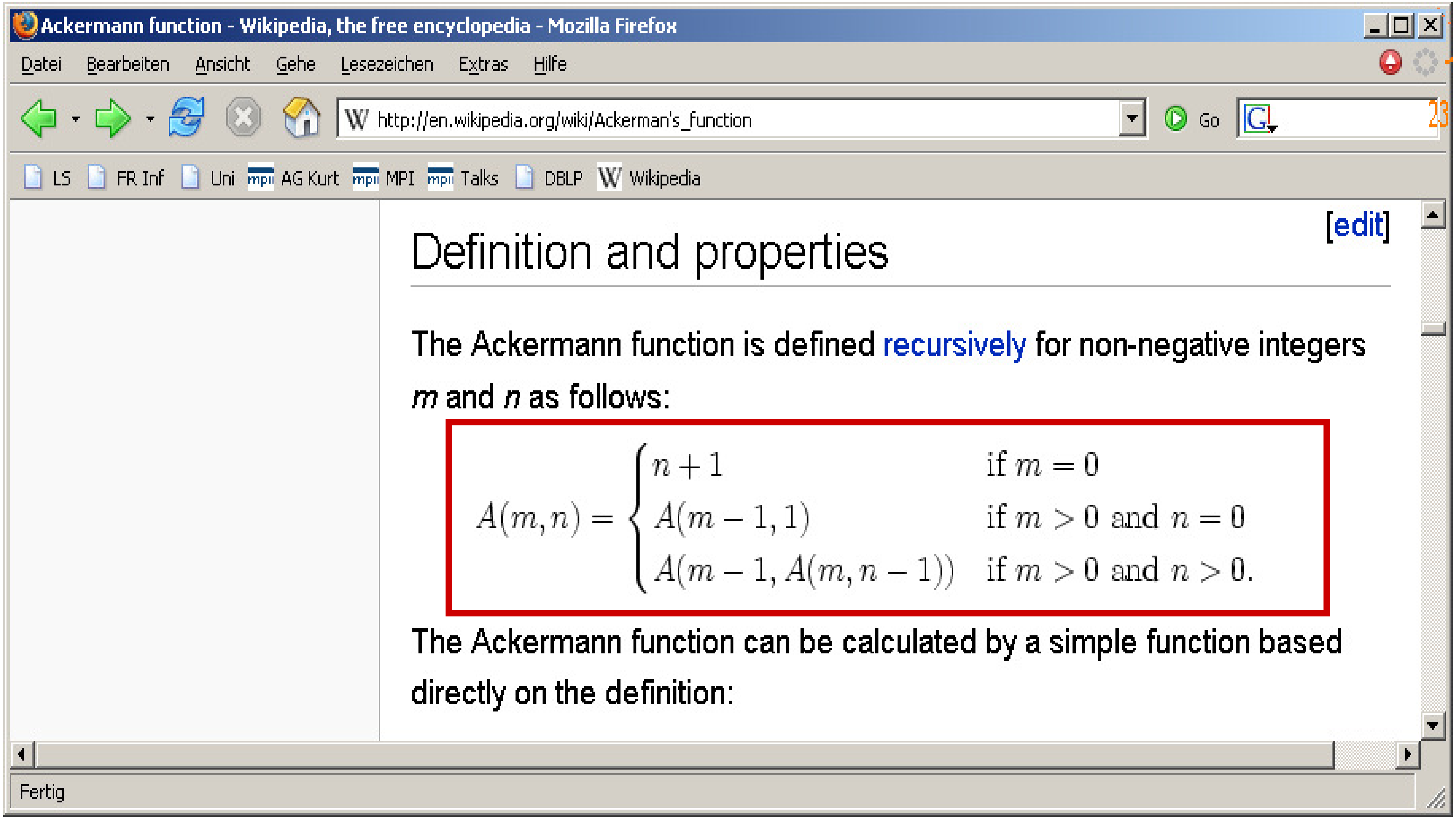
What is this $\alpha(m,n)$??



A two-parameter variation of the inverse Ackermann function can be defined as follows:

$$\boxed{\alpha(m,n) = \min\{i \geq 1 : A(i, \lfloor m/n \rfloor) \geq \log_2 n\}.}$$

This function arises in more precise analyses of the algorithms mentioned above, and gives a more refined time bound. In the disjoint-set data structure, *m* represents the number of operations while *n* represents the number of elements; in the minimum spanning tree algorithm, *m* represents the number of edges while *n* represents the number of vertices. Several slightly different definitions of α(*m*, *n*) exist; for example, $\log_2 n$ is sometimes replaced by *n*, and the floor function is sometimes replaced by a ceiling.



Definition and properties

The Ackermann function is defined recursively for non-negative integers *m* and *n* as follows:

$$A(m,n) = \begin{cases} n+1 & \text{if } m = 0 \\ A(m-1, 1) & \text{if } m > 0 \text{ and } n = 0 \\ A(m-1, A(m, n-1)) & \text{if } m > 0 \text{ and } n > 0. \end{cases}$$

The Ackermann function can be calculated by a simple function based directly on the definition:

This definition of $\alpha(m,n)$
is not particularly enlightening.

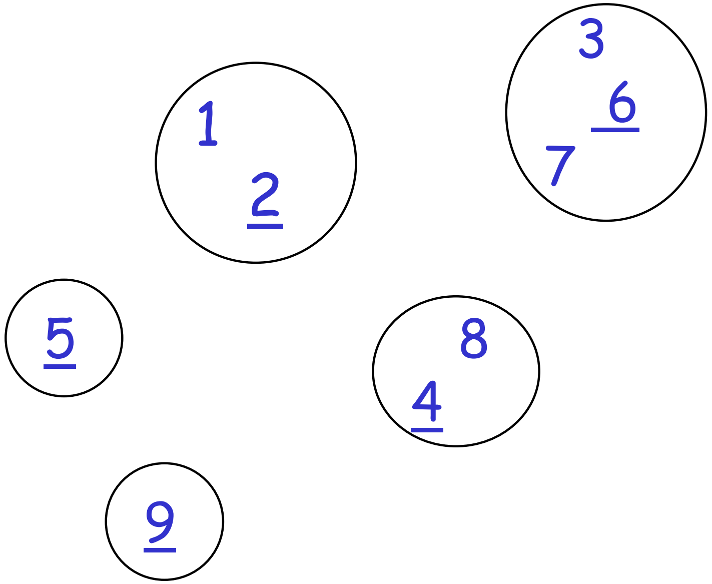
Why does this $\alpha(m,n)$ appear in the analysis of path compression ??

Union Find with Path Compressions

Union Find with Path Compressions

Maintain partition of $S = \{ 1,2,\cdots,n\}$

under operations

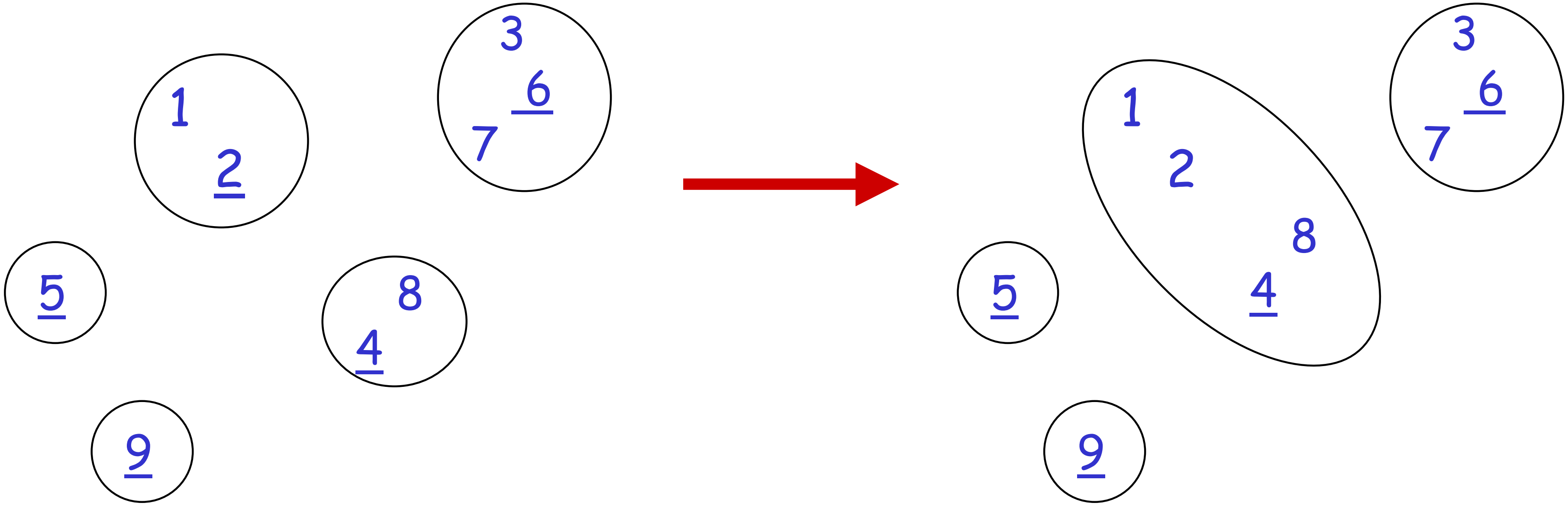


Union Find with Path Compressions

Maintain partition of $S = \{ 1,2,\cdots,n\}$

under operations

Union(2 , 4)

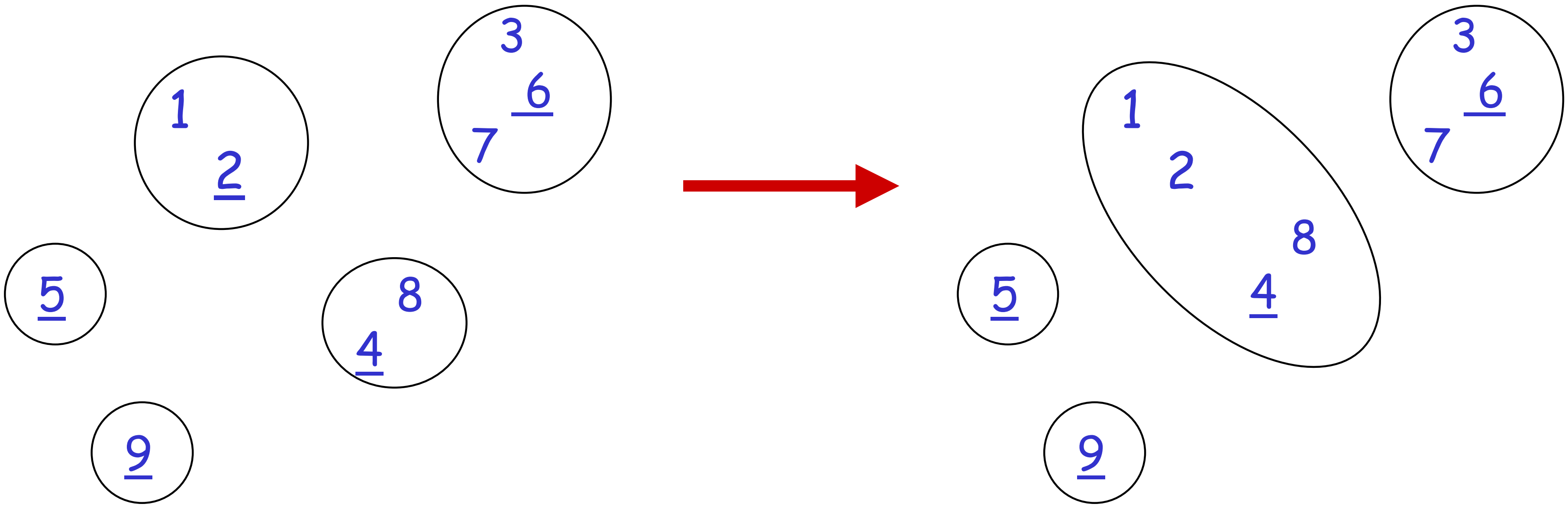


Union Find with Path Compressions

Maintain partition of S = { 1,2,⋯,n}

under operations

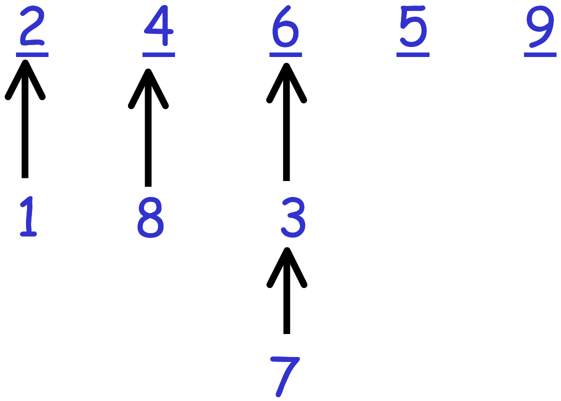
Union(2 , 4)



Find(3) = 6 (representative element)

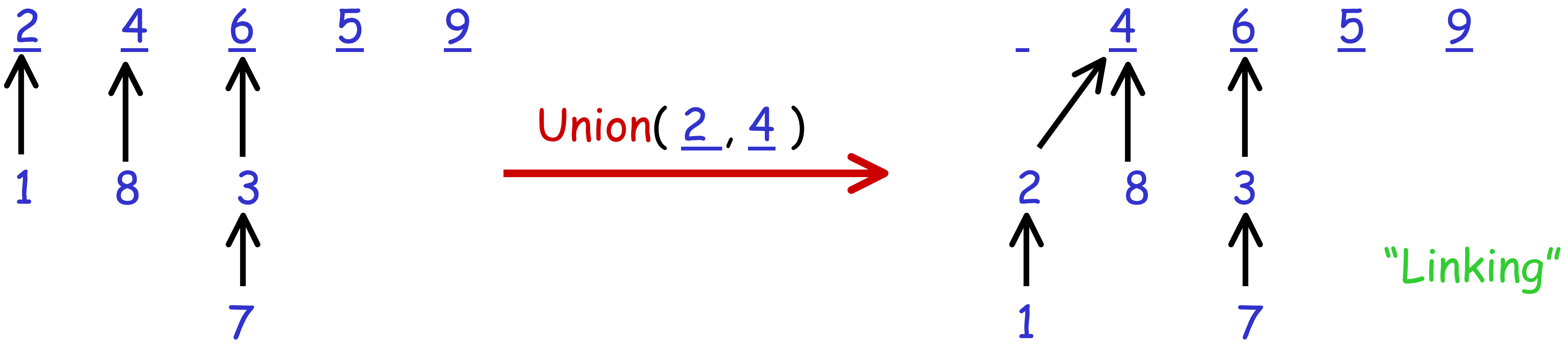
Implementation

* forest $\mathcal{F}$ of rooted trees with node set S
* one tree for each group in current partition
* root of tree is representative of the group



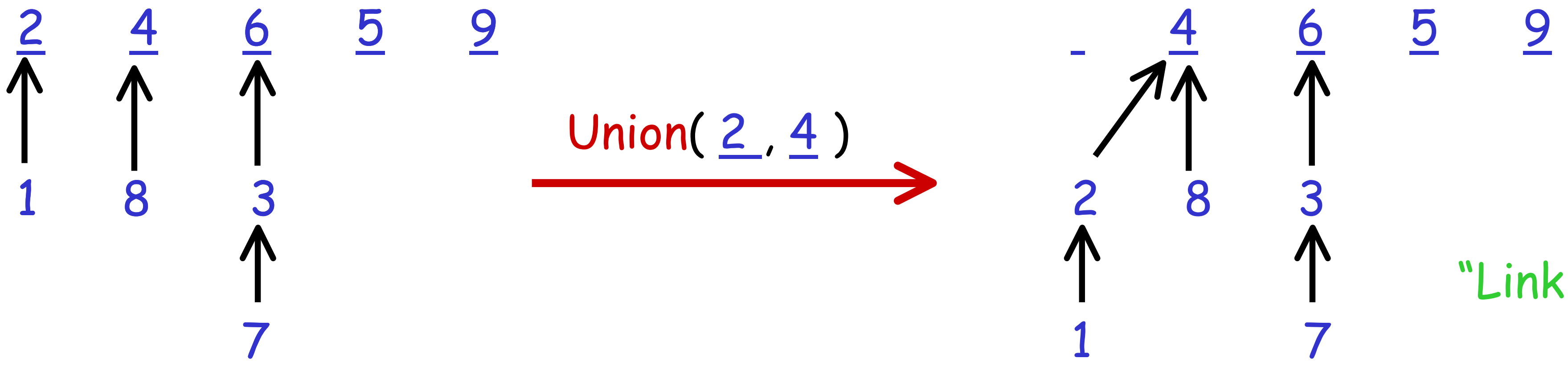
Implementation

- forest $\mathcal{F}$ of rooted trees with node set S
- one tree for each group in current partition
- root of tree is representative of the group



Implementation

* forest $\mathcal{F}$ of rooted trees with node set S
* one tree for each group in current partition
* root of tree is representative of the group

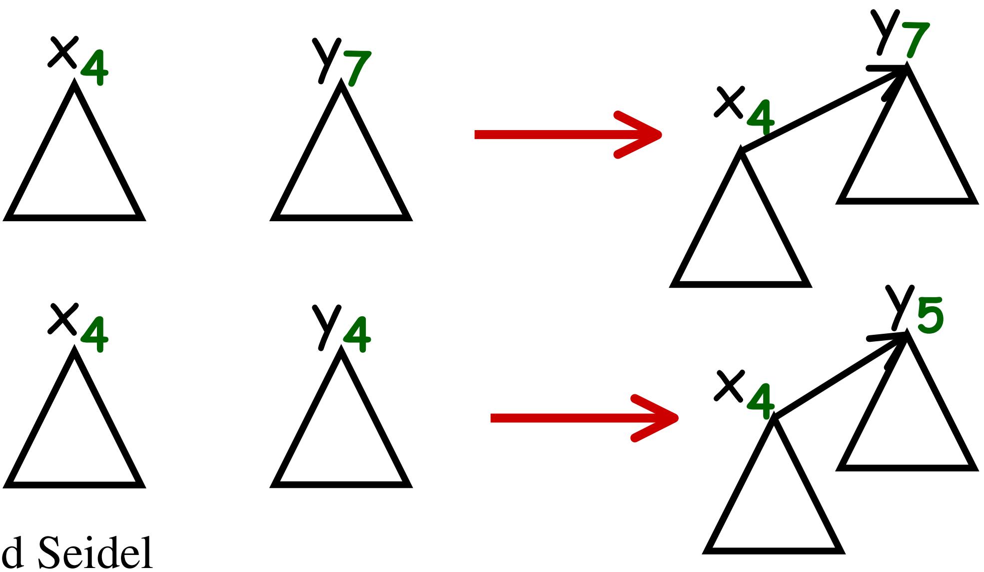


“Linking”

Find(x)   follow path from x to root   “path follwoing”

Heuristic 1: "linking by rank"

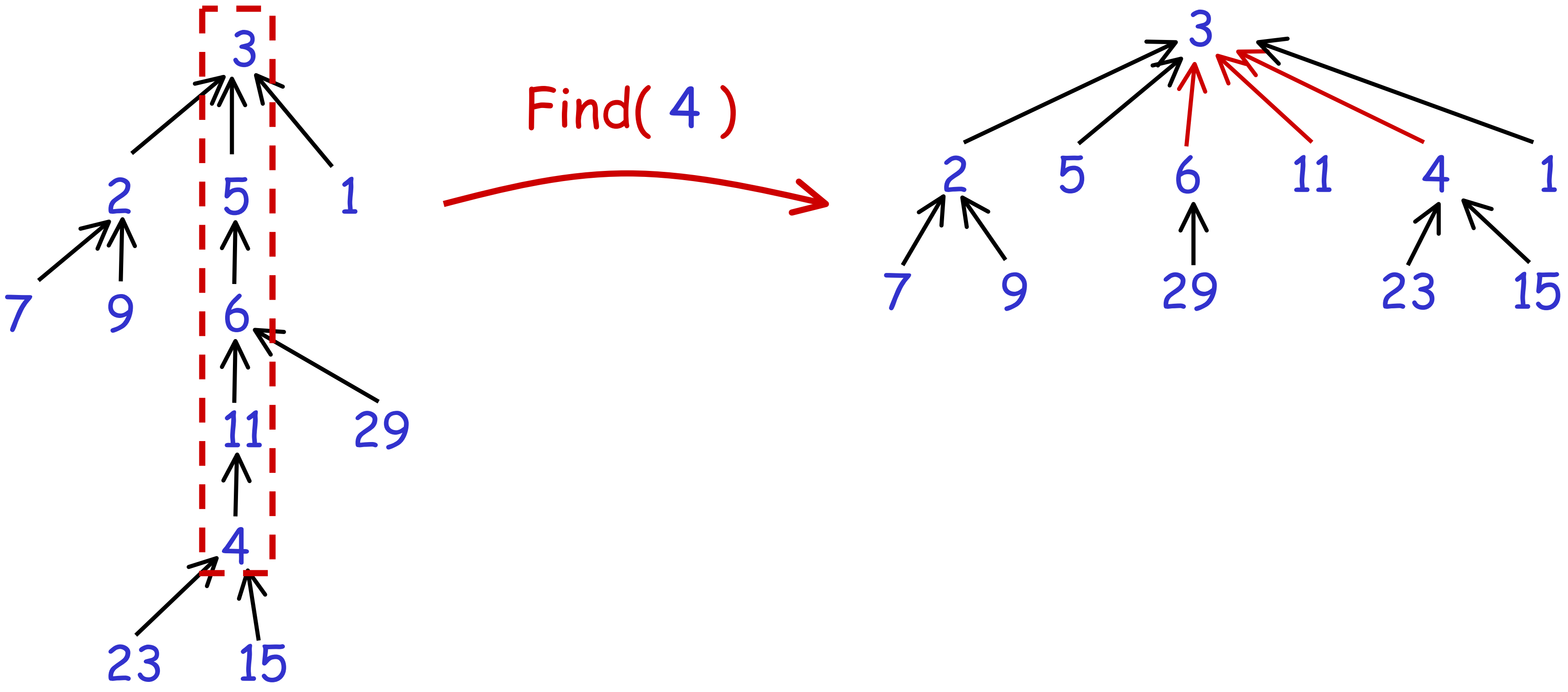
- each node x carries integer rk(x)
- initially rk(x) = 0
- as soon as x is NOT a root, rk(x) stays unchanged
- for Union(x , y) make node with smaller rank child of the other
  in case of tie, increment one of the ranks



© Raimund Seidel

Heuristic 2: Path compression

when performin a Find(x) operation make all nodes in the “findpath” children of the root



SAARLAND UNIVERSITY
COMPUTER SCIENCE

sequence of Union and Find operation

Explicit cost model:

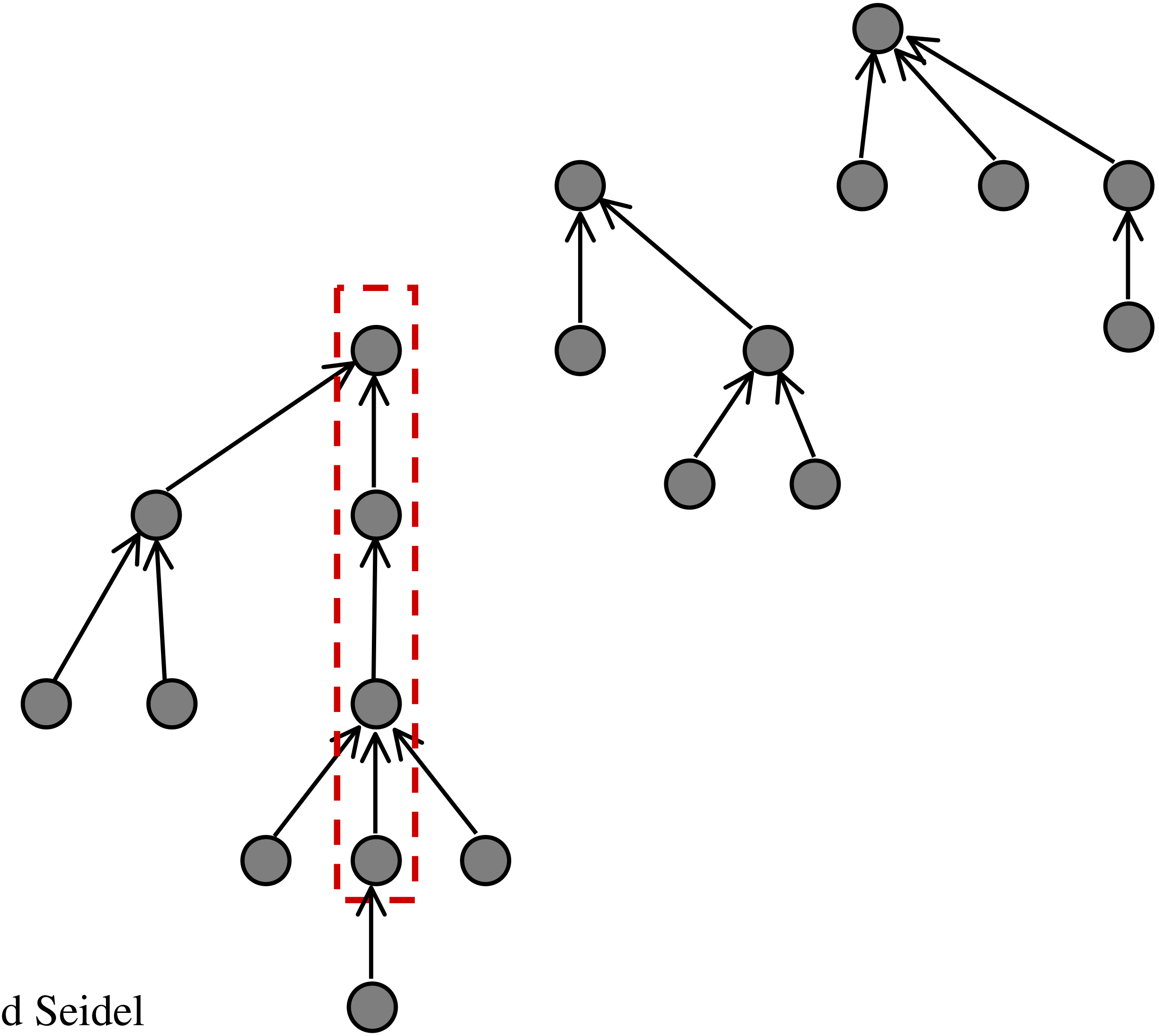
cost(op) = # times some node gets a new parent

Time for Union(x , y) = O(1) = O(cost(Union(x,y)))

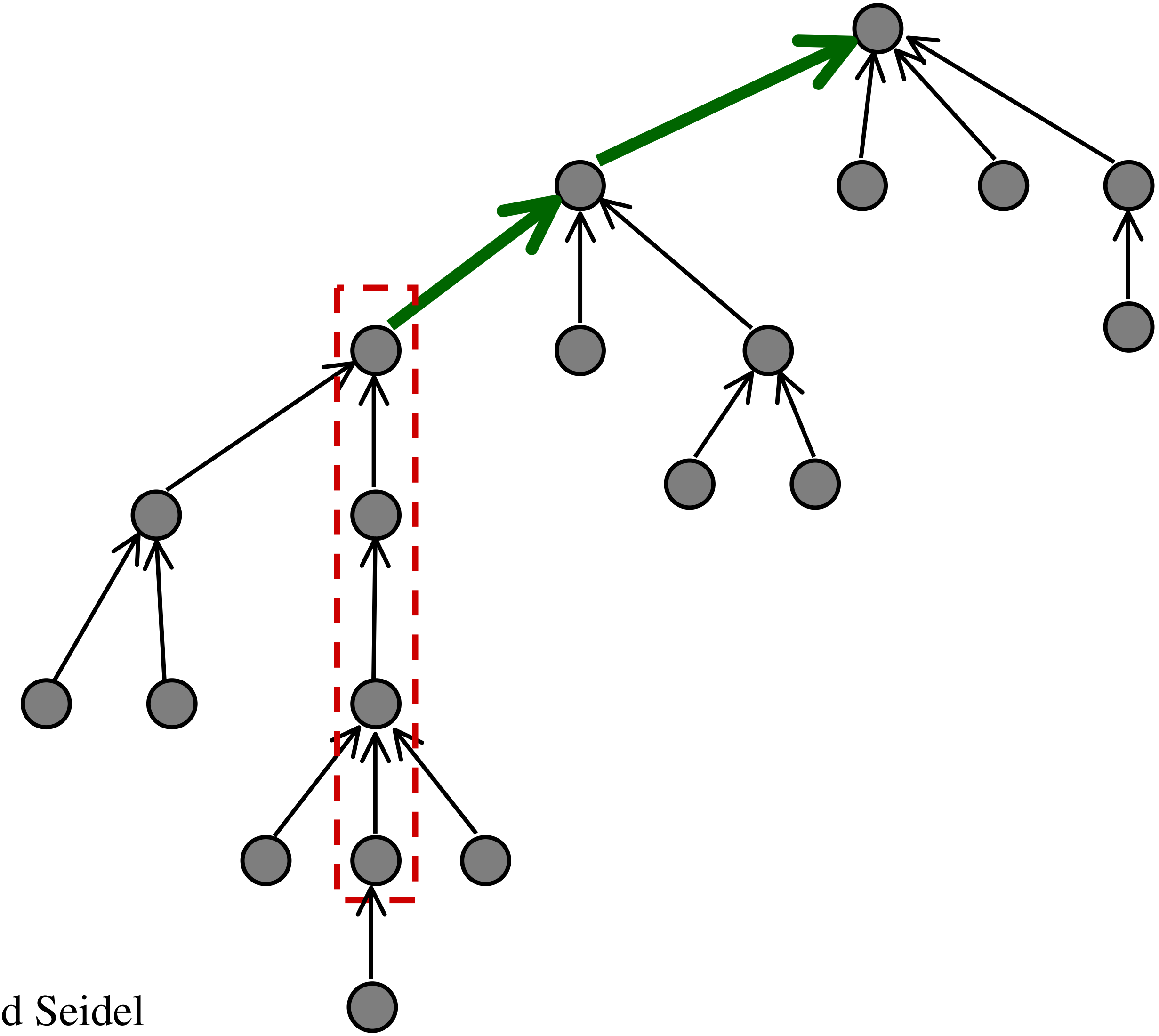
Time for Find(x) = O(# of nodes on findpath)

= O(2 + cost(Find(x)))

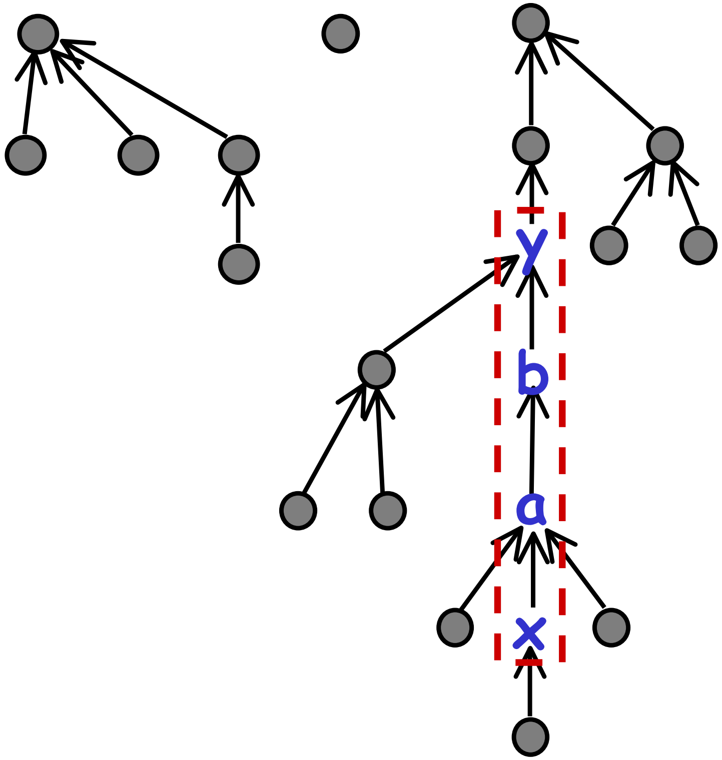
For analysis assume all Unions are performed first, but Find-paths are only followed (and compressed) to correct node.



For analysis assume all Unions are performed first, but Find-paths are only followed (and compressed) to correct node.

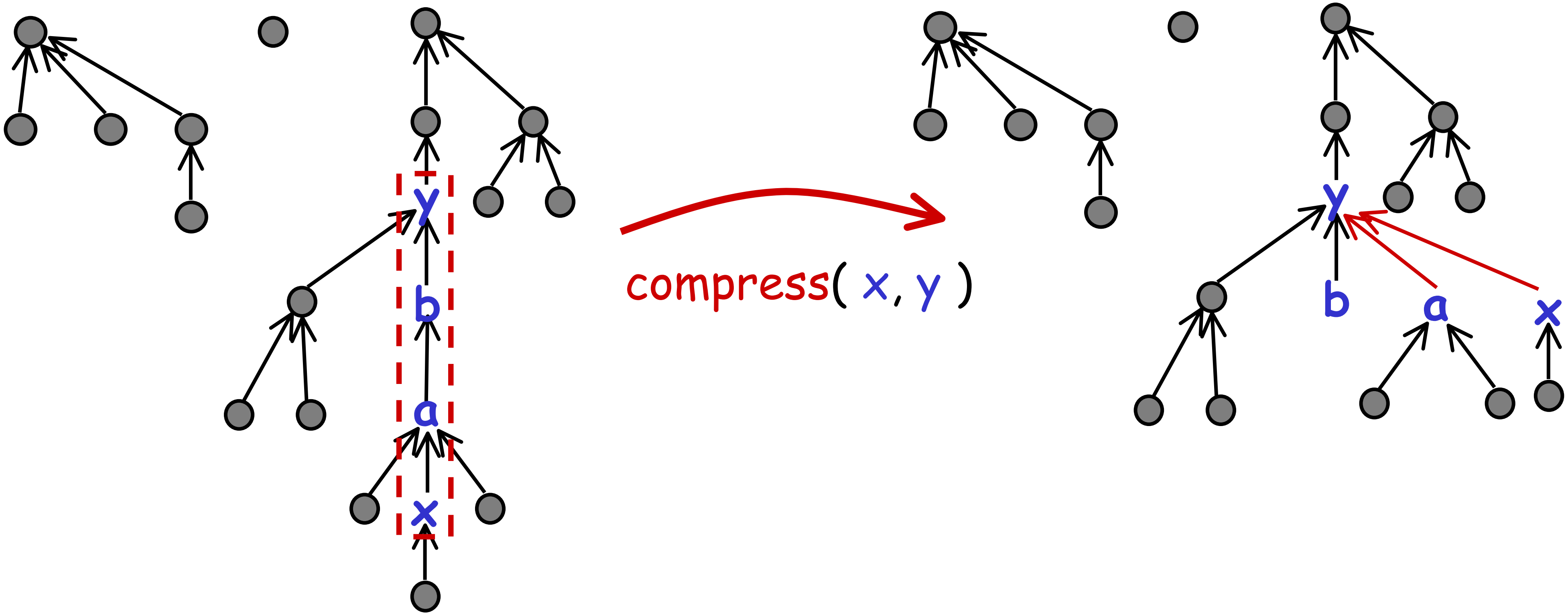


General path compression in forest $\mathcal{F}$

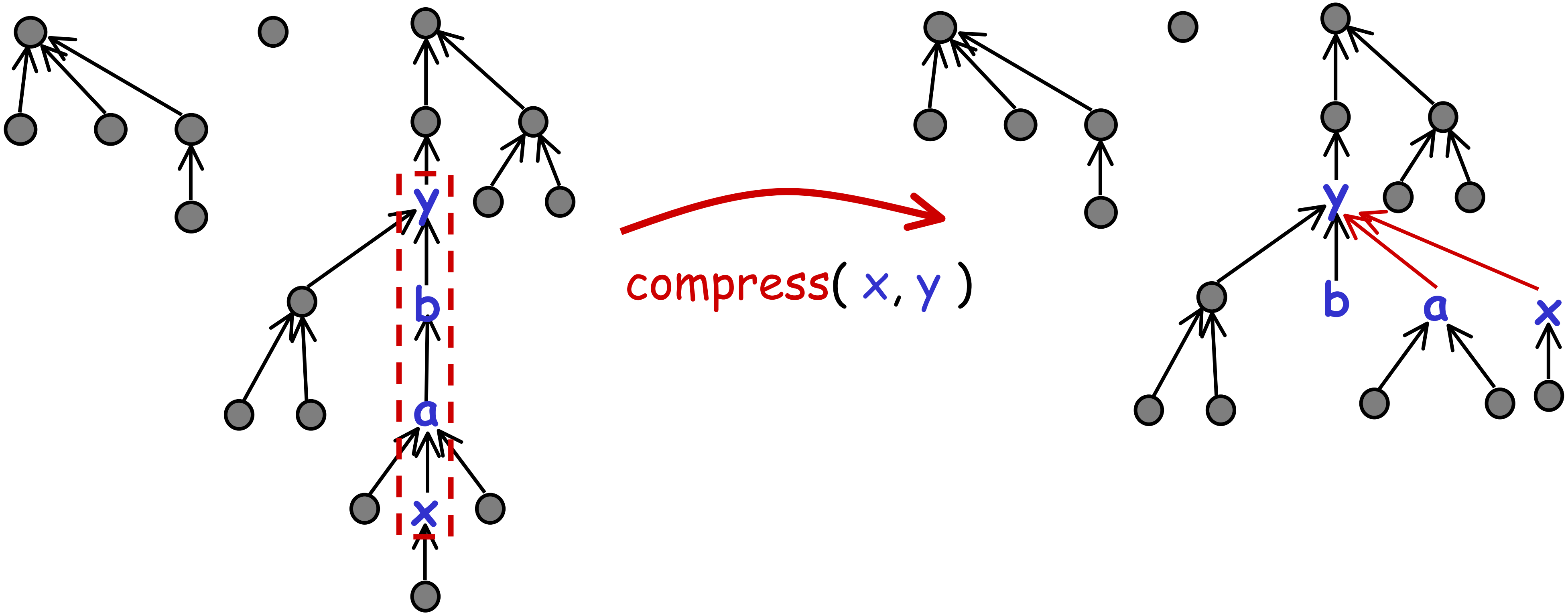


compress(x, y)

General path compression in forest $\mathcal{F}$



General path compression in forest $\mathcal{F}$



cost(compress(x,y)) = # of nodes that get a new parent

Problem formulation

$\mathcal{F}$ forest on node set X

C sequence of compress operations on $\mathcal{F}$

$|C|$ = # of true compress operations in C

cost(C) = $\sum$(cost of individual operations)

How large can cost(C) be at most,
in terms of |X| and |C| ?

Idea:

For the analysis try "*divide and conquer.*"

Idea:

For the analysis try "*divide and conquer.*"

Question:

How do you "*divide*"?

Dissection of a forest $\mathcal{F}$ with node set X :

partition of X into "top part" X_t
and "bottom part" X_b

so that top part X_t is "upwards closed",

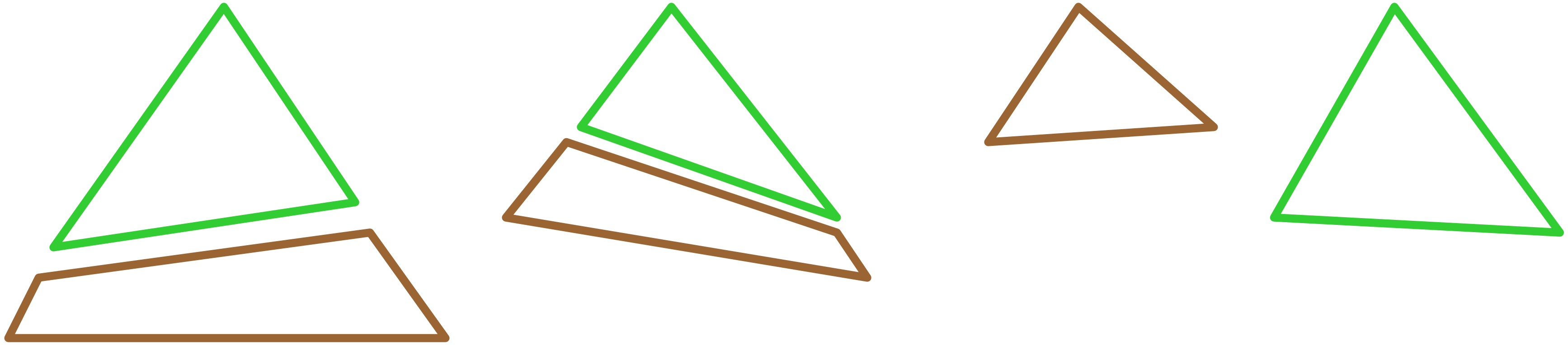
i.e. $x \in X_t \Rightarrow$ every ancestor of x is in X_t also

Dissection of a forest $\mathcal{F}$ with node set X :

partition of X into "top part" X_t
and "bottom part" X_b

so that top part X_t is "upwards closed",

i.e. $x \in X_t \Rightarrow$ every ancestor of x is in X_t also

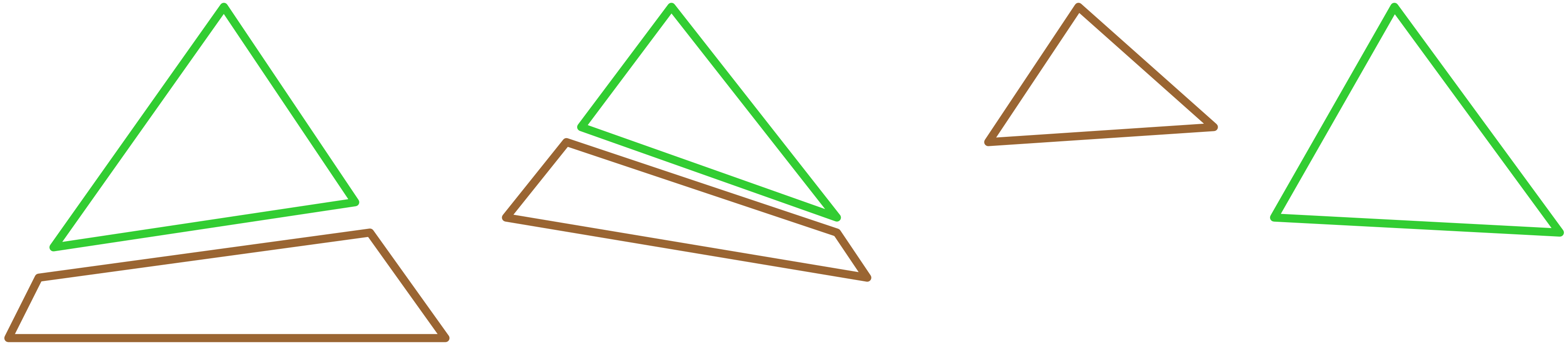


Dissection of a forest $\mathcal{F}$ with node set X :

partition of X into "top part" X_t
and "bottom part" X_b

so that top part X_t is "upwards closed",

i.e. $x \in X_t \Rightarrow$ every ancestor of x is in X_t also



Note: X_t, X_b dissection for $\mathcal{F}$
$\mathcal{F}'$ obtained from $\mathcal{F}$ by
sequence of path compressions

$\Rightarrow$ X_t, X_b is dissection for $\mathcal{F}'$

Main Lemma:

$\mathcal{C}$ ... sequence of operations on $\mathcal{F}$ with node set X

X_t, X_b dissection for $\mathcal{F}$ inducing subforests $\mathcal{F}_t$, $\mathcal{F}_b$

Main Lemma:

$\mathcal{C}$ ... sequence of operations on $\mathcal{F}$ with node set X
X_t, X_b dissection for $\mathcal{F}$ inducing subforests $\mathcal{F}_t$, $\mathcal{F}_b$

$\Rightarrow$ $\exists$ compression sequences
$\mathcal{C}_b$ for $\mathcal{F}_b$ and $\mathcal{C}_t$ for $\mathcal{F}_t$
with

$$\boxed{|\mathcal{C}_b| + |\mathcal{C}_t| \leq |\mathcal{C}|}$$

and

$$\boxed{\text{cost}(\ \mathcal{C}\ ) \leq \text{cost}(\ \mathcal{C}_b\ ) + \text{cost}(\ \mathcal{C}_t\ ) + |X_b| + |\mathcal{C}_t|}$$

Proof: 1) How to get C_b and C_t from C:

Proof: 1) How to get C_b and C_t from C:

compression paths from C

case 1:

into C_t

Proof: 1) How to get C_b and C_t from C:

compression paths from C

case 1:

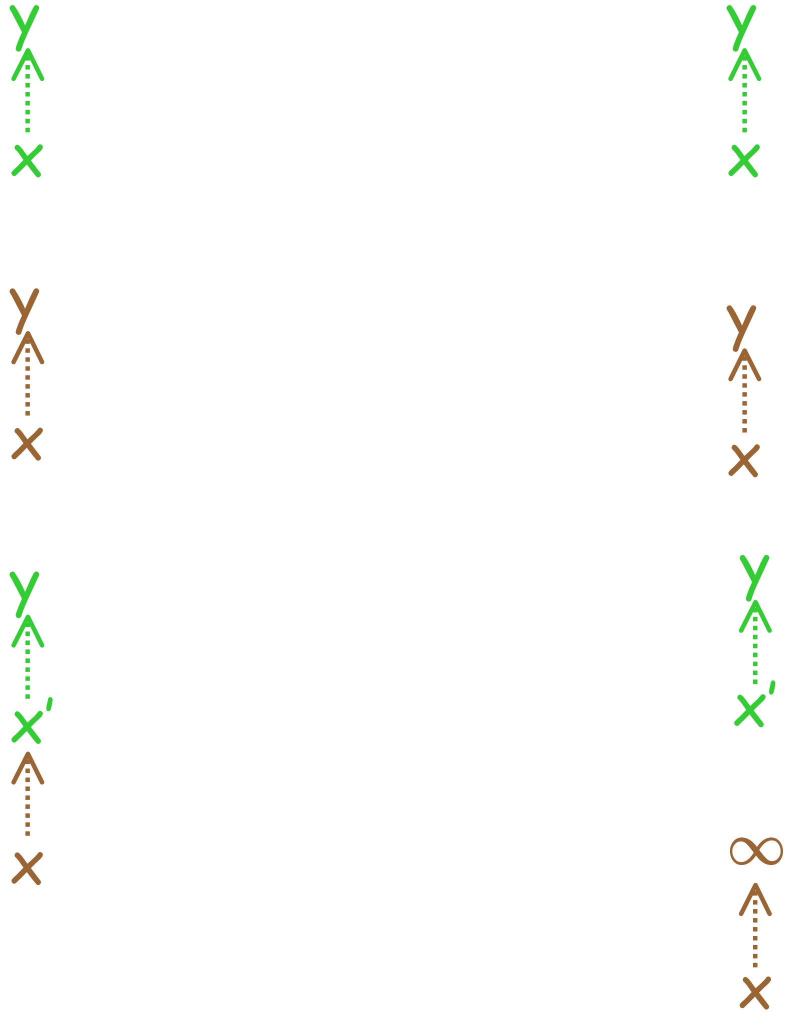
into C_t

case 2:

into C_b

Proof: 1) How to get C_b and C_t from C:

compression paths from C

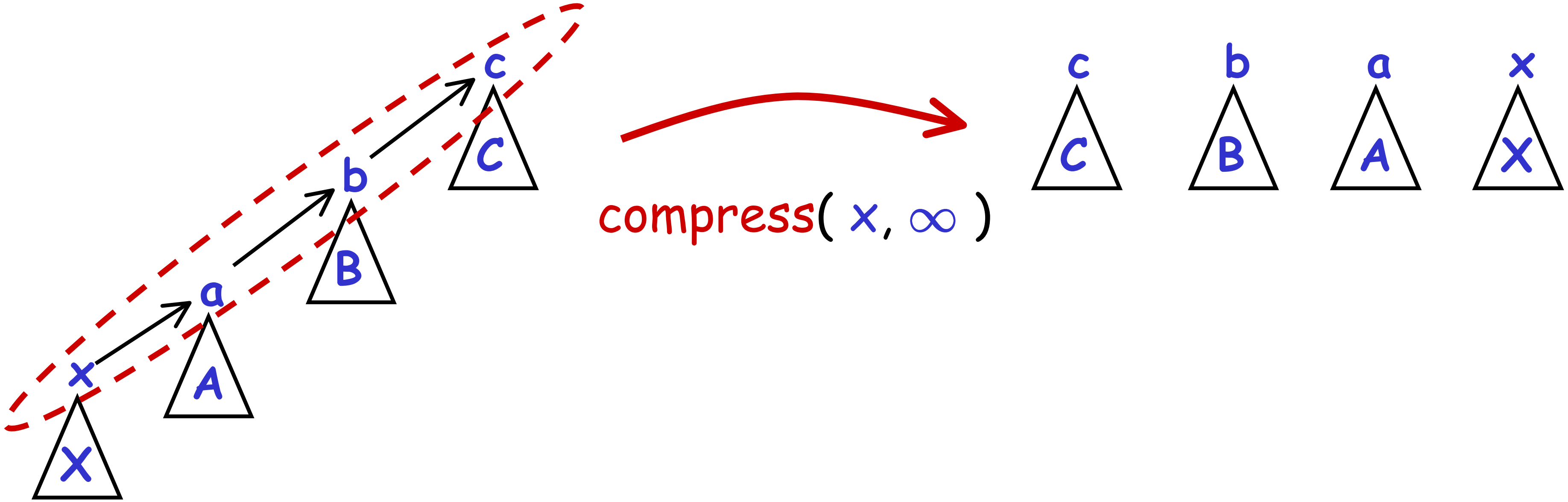


case 1: into C_t

case 2: into C_b

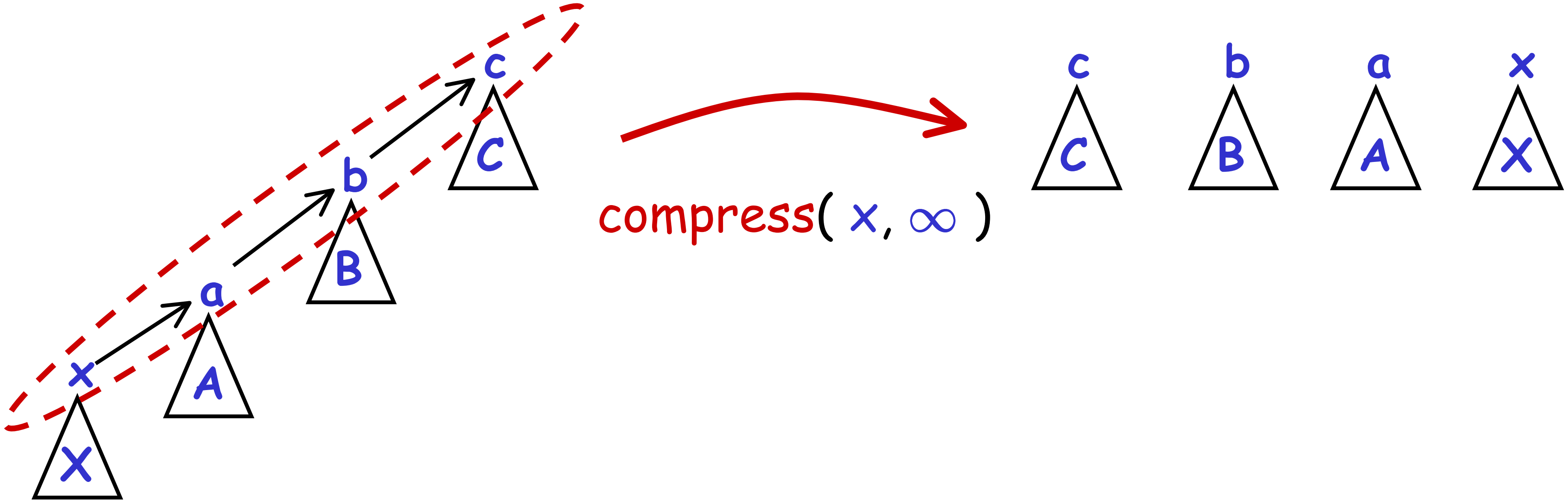
case 3: into C_t, into C_b

"rootpath compress"



compress(x, ∞)

"rootpath compress"



cost(compress(x, ∞)) = # of nodes that get a new parent

= 0

Proof:

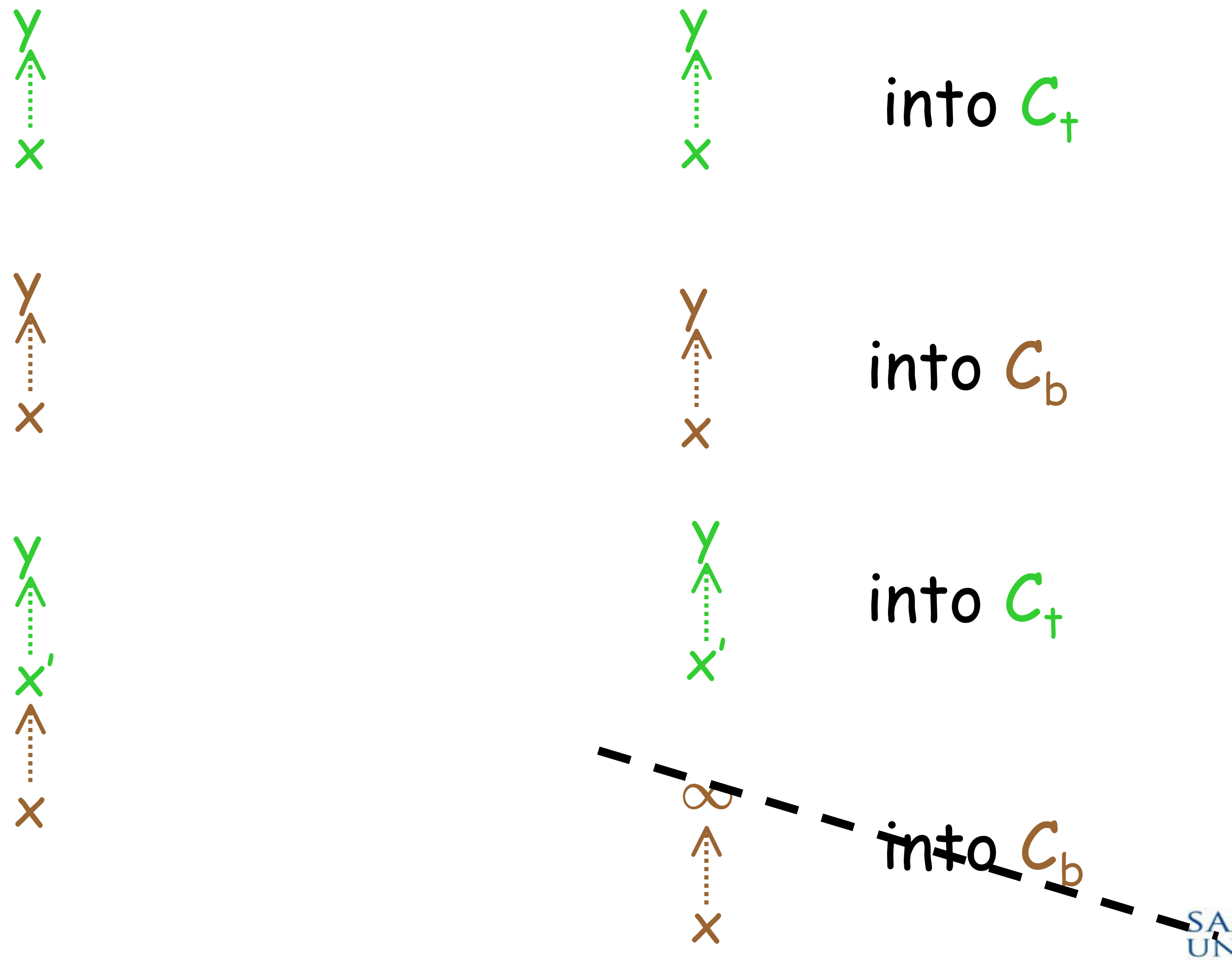
$$\boxed{|C_b| + |C_t| \leq |C|}$$

compression paths from C

case 1:

case 2:

case 3:



into C_t

into C_b

into C_t

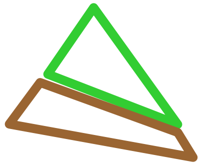
~~into C_b~~

© Raimund Seidel

$$\text{cost}(\,C\,) \leq \text{cost}(\,C_b\,) + \text{cost}(\,C_t\,) + |X_b| + |C_t|$$

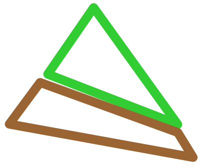
$$\boxed{\mathrm{cost}(\ C\ ) \le \mathrm{cost}(\ C_b\ ) + \mathrm{cost}(\ C_t\ ) + |X_b| + |C_t|}$$

$\mathrm{cost}(\ C\ )$



© Raimund Seidel

$$\boxed{\text{cost}(\ C\ ) \leq \text{cost}(\ C_b\ ) + \text{cost}(\ C_t\ ) + |X_b| + |C_t|}$$

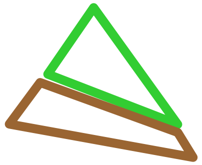


cost(C)

green node gets new green parent:	accounted by cost(C_t)

© Raimund Seidel

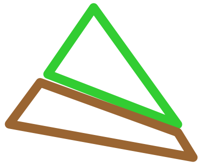
$$\boxed{\text{cost}(\ C\ ) \leq \text{cost}(\ C_b\ ) + \text{cost}(\ C_t\ ) + |X_b| + |C_t|}$$



cost(C)	
green node gets new green parent:	accounted by cost(C_t)
brown node gets new brown parent:	accounted by cost(C_b)

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$\boxed{\mathrm{cost}(\,C\,) \leq \mathrm{cost}(\,C_b\,) + \mathrm{cost}(\,C_t\,) + |X_b| + |C_t|}$$

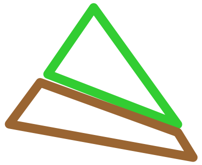


cost(C)	
green node gets new green parent:	accounted by $\mathrm{cost}(C_t)$
brown node gets new brown parent:	accounted by $\mathrm{cost}(C_b)$
brown node gets new green parent: for the first time	accounted by $\lvert X_b\rvert$

© Raimund Seidel

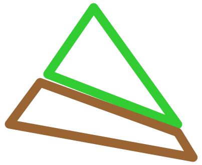
$$\boxed{\text{cost}(\ \mathcal{C}\ ) \le \text{cost}(\ \mathcal{C}_b\ ) + \text{cost}(\ \mathcal{C}_t\ ) + |X_b| + |\mathcal{C}_t|}$$

$\text{cost}(\ \mathcal{C}\ )$



green node gets new green parent:	accounted by $\text{cost}(\mathcal{C}_t)$
brown node gets new brown parent:	accounted by $\text{cost}(\mathcal{C}_b)$
brown node gets new green parent: for the first time	accounted by $|X_b|$ $- \#\text{roots}(\ \mathcal{F}_b\ )$

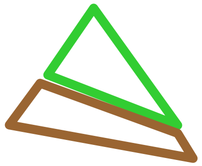
$$\boxed{\text{cost}(\mathcal{C}) \leq \text{cost}(\mathcal{C}_b) + \text{cost}(\mathcal{C}_t) + |X_b| - \#\text{roots}(\mathcal{F}_b) + |\mathcal{C}_t|}$$



cost($\mathcal{C}$)	
green node gets new green parent:	accounted by cost($\mathcal{C}_t$)
brown node gets new brown parent:	accounted by cost($\mathcal{C}_b$)
brown node gets new green parent: for the first time	accounted by $|X_b|$ - #roots($\mathcal{F}_b$)

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$\boxed{\text{cost}(C) \leq \text{cost}(C_b) + \text{cost}(C_t) + |X_b| - \#\text{roots}(\mathcal{F}_b) + |C_t|}$$



cost(C)

green node gets new green parent:	accounted by cost(C_t)
brown node gets new brown parent:	accounted by cost(C_b)
brown node gets new green parent: for the first time	accounted by $\lvert X_b\rvert$ - #roots($\mathcal{F}_b$)
brown node gets new green parent: again	accounted by $\lvert C_t\rvert$

Main Lemma':

C ... sequence of operations on $\mathcal{F}$ with node set X

X_t, X_b dissection for $\mathcal{F}$ inducing subforests $\mathcal{F}_t$, $\mathcal{F}_b$

$\Rightarrow$ $\exists$ compression sequences
C_b for $\mathcal{F}_b$ and C_t for $\mathcal{F}_t$
with

$$\boxed{|C_b| + |C_t| \leq |C|}$$

and

$$\boxed{\begin{aligned} \text{cost}(\,C\,) \leq \text{cost}(\,C_b\,) + \text{cost}(\,C_t\,) \\ + |X_b| - \#\text{roots}(\mathcal{F}_b) + |C_t| \end{aligned}}$$

$f(m,n)$ ... maximum cost of any compression sequence C with $|C|=m$ in an arbitrary forest with n nodes.

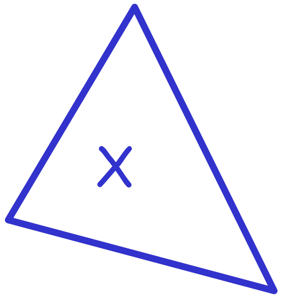
Claim: $f(m,n) \leq (m+n)\cdot\log_2 n$

Claim: $f(m,n) \leq (m+n) \cdot \log_2 n$

Claim: $f(m,n) \leq (m+n) \cdot \log_2 n$

Proof:

forest $\mathcal{F}$



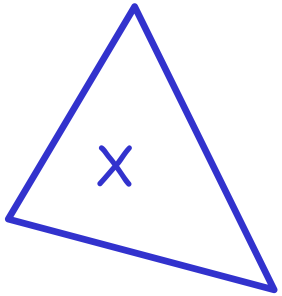
$|X|=n$

C compression sequence $|C|=m$

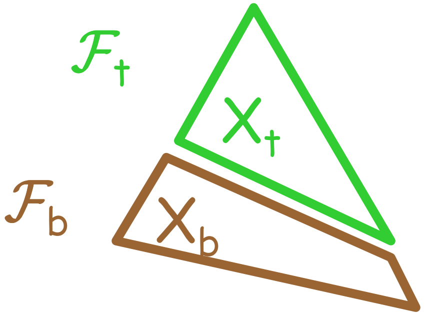
Claim: $f(m,n) \leq (m+n)\cdot\log_2 n$

Proof:

forest $\mathcal{F}$



$|X|=n$



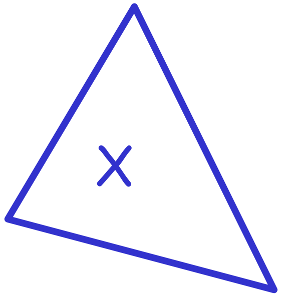
$|X_t|=|X_b|=n/2$

C compression sequence $|C|=m$

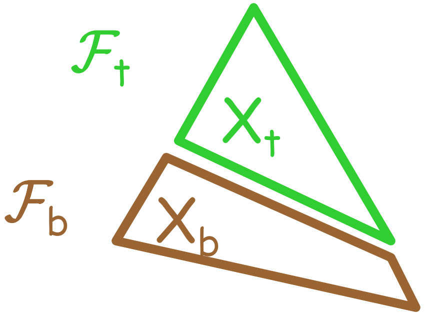
Claim: $f(m,n) \leq (m+n)\cdot\log_2 n$

Proof:

forest $\mathcal{F}$



$|X|=n$



$|X_t|=|X_b|=n/2$

C compression sequence $|C|=m$

Main Lemma $\Rightarrow\ \exists\ C_t, C_b \quad |C_b|+|C_t| \leq |C|$

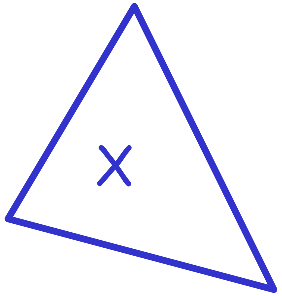
$m_b + m_t \leq m$

$$\text{cost}(C) \leq \text{cost}(C_b) + \text{cost}(C_t) + |X_b| + |C_t|$$

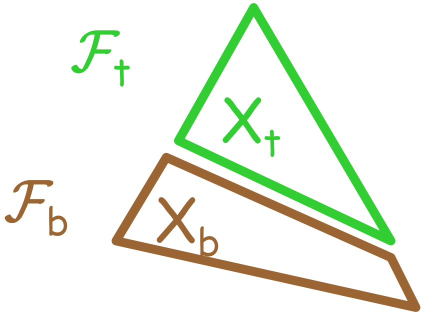
Claim: $f(m,n) \le (m+n)\cdot\log_2 n$

Proof:

forest $\mathcal{F}$



$|X|=n$



$|X_t|=|X_b|=n/2$

$\mathcal{C}$ compression sequence $|\mathcal{C}|=m$

Main Lemma $\Rightarrow$ $\exists\ \mathcal{C}_t, \mathcal{C}_b$ $|\mathcal{C}_b|+|\mathcal{C}_t| \le |\mathcal{C}|$

$m_b + m_t \le m$

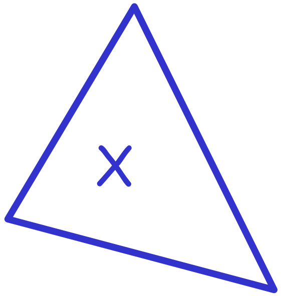
$$\text{cost}(\mathcal{C}) \le \text{cost}(\mathcal{C}_b) + \text{cost}(\mathcal{C}_t) + |X_b| + |\mathcal{C}_t|$$

Induction: $\le (m_b+n/2)\log n/2 + (m_t+n/2)\log n/2 + n/2 + m_t$

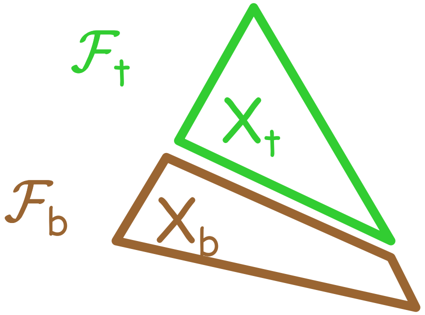
Claim: $f(m,n) \leq (m+n)\cdot\log_2 n$

Proof:

forest $\mathcal{F}$



$|X|=n$



$|X_t|=|X_b|=n/2$

C compression sequence $|C|=m$

Main Lemma $\Rightarrow \exists\ C_t, C_b \quad |C_b|+|C_t| \leq |C|$

$$m_b + m_t \leq m$$

$$\text{cost}(C) \leq \text{cost}(C_b) + \text{cost}(C_t) + |X_b| + |C_t|$$

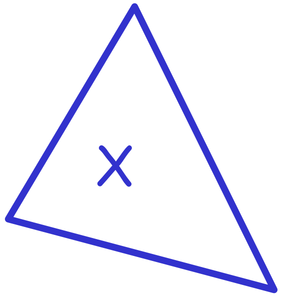
Induction: $\leq (m_b+n/2)\log n/2 + (m_t+n/2)\log n/2 + n/2 + m_t$

$$\leq (m_b+m_t+n/2+n/2)\log n/2 + n + m$$

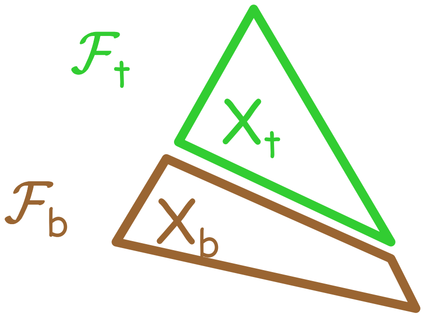
Claim: $f(m,n) \leq (m+n)\cdot\log_2 n$

Proof:

forest $\mathcal{F}$



$|X|=n$



$|X_t|=|X_b|=n/2$

C compression sequence $|C|=m$

Main Lemma $\Rightarrow \exists\, C_t, C_b \quad |C_b|+|C_t| \leq |C|$

$m_b + m_t \leq m$

$$\text{cost}(C) \leq \text{cost}(C_b) + \text{cost}(C_t) + |X_b| + |C_t|$$

Induction: $\leq (m_b+n/2)\log n/2 + (m_t+n/2)\log n/2 + n/2 + m_t$

$$\leq (m_b+m_t+n/2+n/2)\log n/2 + n + m$$

$$\leq (m+n)\cdot\log_2 n/2 + (m+n) = (m+n)\cdot\log_2 n$$

Corollary:

Any sequence of m Union, Find operations in a universe of n elements that uses arbitrary linking and path compression takes time at most

$$O(\,(m+n)\cdot\log n)$$

Corollary:

Any sequence of m Union, Find operations in a universe of n elements that uses arbitrary linking and path compression takes time at most

$$O(\ (m+n)\cdot\log n)$$

By choosing a dissection that is "unbalanced" in relation to m/n one can prove a better bound of

$$O(\ (m+n)\cdot\log_{\lceil m/n\rceil+1} n)$$

Corollary:

Any sequence of m Union, Find operations in a universe of n elements that uses arbitrary linking and path compression takes time at most

$$O(\ (m+n)\cdot \log n)$$

By choosing a dissection that is "unbalanced" in relation to m/n one can prove a better bound of

$$O(\ (m+n)\cdot \log_{\lceil m/n\rceil+1} n)$$

Proof: exercise

Path compression and union by rank

$f : \mathbb{N} \rightarrow \mathbb{R}$

Brief digression

$f : \mathbb{N} \to \mathbb{R}$

Brief digression

$$f^*(n) = \begin{cases} 0 & \text{if } n \leq 1 \\ 1 + f^*(\, f(n)\,) & \text{if } n > 1 \end{cases}$$

$f : \mathbb{N} \to \mathbb{R}$

Brief digression

$$f^*(n) = \begin{cases} 0 & \text{if } n \leq 1 \\ 1 + f^*(\,f(n)\,) & \text{if } n > 1 \end{cases}$$

$$f^*(n) = \min \{\, k \mid \underbrace{f(f(\,\cdots\cdots\, f}_{k \text{ times}}(n)\cdots) \leq 1 \,\}$$

$f : \mathbb{N} \to \mathbb{R}$

Brief digression

$$f^*(n) = \begin{cases} 0 & \text{if } n \leq 1 \\ 1 + f^*(\,f(n)\,) & \text{if } n > 1 \end{cases}$$

$$f^*(n) = \min\{\, k \mid \underbrace{f(f(\,\cdots\cdots\, f}_{k \text{ times}}(n)\cdots) \leq 1 \,\}$$

Properties: f a "nice" compaction, i.e. $f(n)<n-1$

$\Rightarrow$ f^* a "nice" compaction and

f^* "much smaller" than f

Examples for f^* :

$f(n)$	$f^*(n)$
$n-1$	$n-1$
$n-2$	$n/2$
$n-c$	n/c
$n/2$	$\log_2 n$
n/c	$\log_c n$
$\sqrt{n}$	$\log \log n$
$\log n$	$\log^* n$

Path compression and union by rank

Def: $\mathcal{F}$ forest, x node in $\mathcal{F}$

$r(x)$ = height of subtree rooted at x

($r(\text{leaf}) = 0$)

$\mathcal{F}$ is a **rank forest**, if

for every node x

for every i with $0 \leq i < r(x)$,

there is a child y_i of x with $r(y_i)=i$.

© Raimund Seidel

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Path compression and union by rank

Def: $\mathcal{F}$ forest, x node in $\mathcal{F}$

$r(x)$ = height of subtree rooted at x

($r(\text{leaf}) = 0$)

$\mathcal{F}$ is a **rank forest**, if

for every node x

for every i with $0 \le i < r(x)$,

there is a child y_i of x with $r(y_i)=i$.

Note: Union by rank produces rank forests !

Path compression and union by rank

Def: $\mathcal{F}$ forest, x node in $\mathcal{F}$

$r(x)$ = height of subtree rooted at x

($r(\text{leaf}) = 0$)

$\mathcal{F}$ is a **rank forest**, if

for every node x

for every i with $0 \leq i < r(x)$,

there is a child y_i of x with $r(y_i)=i$.

Note: Union by rank produces rank forests !

Lemma: $r(x)=r \Rightarrow x$ has at least r children.

Path compression and union by rank

Def: $\mathcal{F}$ forest, x node in $\mathcal{F}$

$r(x)$ = height of subtree rooted at x

($r(\text{leaf}) = 0$)

$\mathcal{F}$ is a **rank forest**, if

for every node x

for every i with $0 \leq i < r(x)$,

there is a child y_i of x with $r(y_i)=i$.

Note: Union by rank produces rank forests !

Lemma: $r(x)=r \Rightarrow x$ has at least r children

and at least 2^r descendants.

Inheritance Lemma:

$\mathcal{F}$ rank forest with maximum rank r and node set X

$s \in \mathbb{N}$: $X_{>s} = \{ x \in X \mid r(x) > s \}$ $\mathcal{F}_{>s}$

$X_{\leq s} = \{ x \in X \mid r(x) \leq s \}$ $\mathcal{F}_{\leq s}$ induced forests

Inheritance Lemma:

$\mathcal{F}$ rank forest with maximum rank r and node set X

$s \in \mathbb{N}$: $X_{>s} = \{ x \in X \mid r(x) > s \}$ $\mathcal{F}_{>s}$ induced forests

$X_{\leq s} = \{ x \in X \mid r(x) \leq s \}$ $\mathcal{F}_{\leq s}$

i) $X_{\leq s}$, $X_{>s}$ is a dissection for $\mathcal{F}$

ii) $\mathcal{F}_{\leq s}$ is a rank forest with maximum rank $\leq$ s

iii) $\mathcal{F}_{>s}$ is a rank forest with maximum rank $\leq$ r-s-1

Inheritance Lemma:

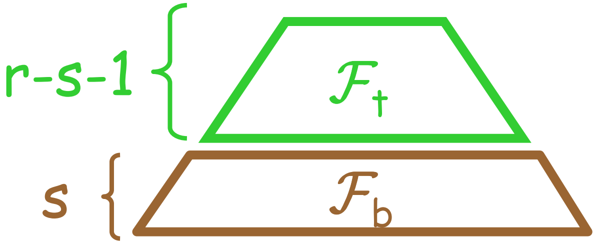
$\mathcal{F}$ rank forest with maximum rank r and node set X

$s \in \mathbb{N}$: $X_{>s} = \{ x \in X \mid r(x) > s \}$ $\mathcal{F}_{>s}$
$X_{\leq s} = \{ x \in X \mid r(x) \leq s \}$ $\mathcal{F}_{\leq s}$ induced forests

i) $X_{\leq s}$, $X_{>s}$ is a dissection for $\mathcal{F}$

ii) $\mathcal{F}_{\leq s}$ is a rank forest with maximum rank $\leq s$

iii) $\mathcal{F}_{>s}$ is a rank forest with maximum rank $\leq r\text{-}s\text{-}1$



Inheritance Lemma:

$\mathcal{F}$ rank forest with maximum rank r and node set X

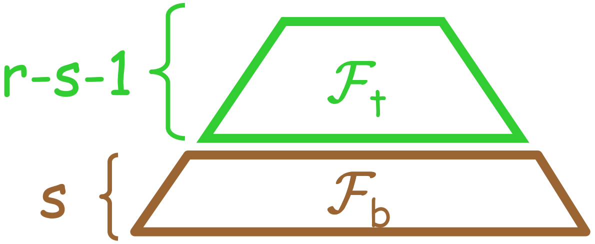
$s \in \mathbb{N}$: $X_{>s} = \{ x \in X \mid r(x) > s \}$ $\mathcal{F}_{>s}$

$X_{\leq s} = \{ x \in X \mid r(x) \leq s \}$ $\mathcal{F}_{\leq s}$ induced forests

i) $X_{\leq s}$, $X_{>s}$ is a dissection for $\mathcal{F}$

ii) $\mathcal{F}_{\leq s}$ is a rank forest with maximum rank $\leq s$

iii) $\mathcal{F}_{>s}$ is a rank forest with maximum rank $\leq r-s-1$



Proofs: exercise

© Raimund Seidel

$f(m,n,r)$ = maximum cost of any compression sequence C, with $|C|=m$, in rank forest $\mathcal{F}$ with n nodes and maximum rank r.

$f(m,n,r)$ = maximum cost of any compression sequence C, with $|C|=m$, in rank forest $\mathcal{F}$ with n nodes and maximum rank r.

Trivial bounds:

$$f(m,n,r) \leq (r-1)\cdot n$$

$$f(m,n,r) \leq (r-1)\cdot m$$

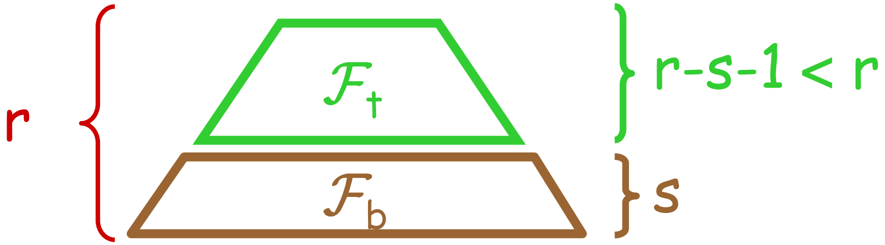
$f(m,n,r)$ = maximum cost of any compression sequence C, with $|C|=m$, in rank forest $\mathcal{F}$ with n nodes and maximum rank r.

Trivial bounds:

$$f(m,n,r) \leq (r-1)\cdot n$$

$$f(m,n,r) \leq (r-1)\cdot m$$

$$f(m,n,r) \leq m + (r-2)\cdot n$$



r

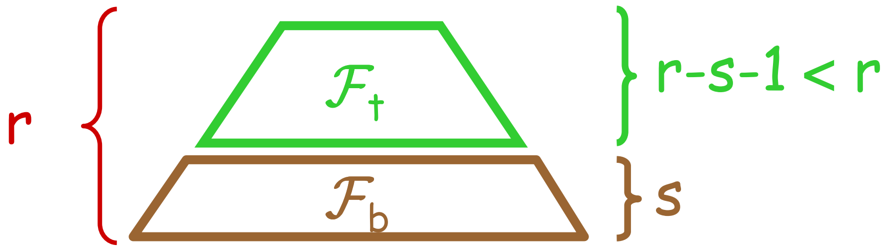
$\mathcal{F}_t$ $\}\ r-s-1 < r$

$\mathcal{F}_b$ $\}\ s$

$|X_{>s}| = n_t$ $\qquad |C_t| = m_t$

$|X_{\leq s}| = n_b = n-n_t$ $\qquad |C_b| = m_b$

$$\text{cost}(\ C\ ) \leq \text{cost}(\ C_t\ ) + \text{cost}(\ C_b\ ) + |X_b| - \#\text{rts}(\mathcal{F}_b) + |C_t|$$

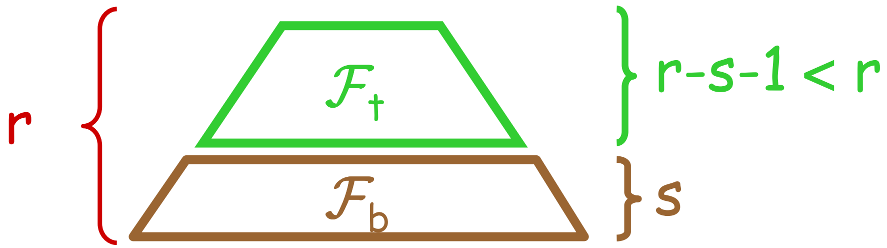


$|X_{>s}| = n_t$ $\quad$ $|\mathcal{C}_t| = m_t$

$|X_{\le s}| = n_b = n-n_t$ $\quad$ $|\mathcal{C}_b| = m_b$

$$\text{cost}(\,\mathcal{C}\,) \le \text{cost}(\,\mathcal{C}_t\,) + \text{cost}(\,\mathcal{C}_b\,) + |X_b| - \#\text{rts}(\mathcal{F}_b) + |\mathcal{C}_t|$$

$$\le f(m_t, n_t, r-s-1) +$$

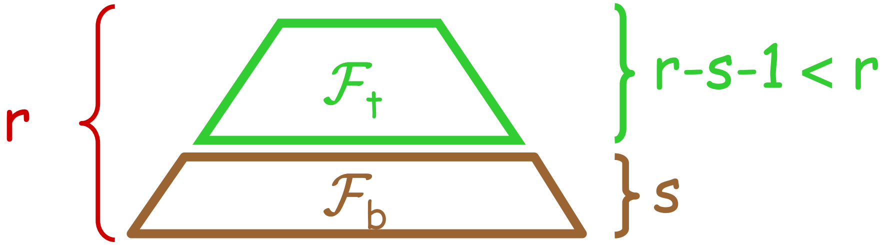


$|X_{>s}| = n_t$ $|\mathcal{C}_t| = m_t$

$|X_{\le s}| = n_b = n-n_t$ $|\mathcal{C}_b| = m_b$

$$\text{cost}(\,\mathcal{C}\,) \le \text{cost}(\,\mathcal{C}_t\,) + \text{cost}(\,\mathcal{C}_b\,) + |X_b| - \#\text{rts}(\mathcal{F}_b) + |\mathcal{C}_t|$$

$$\le f(m_t,n_t,r-s-1) + f(m_b,n_b,s) +$$

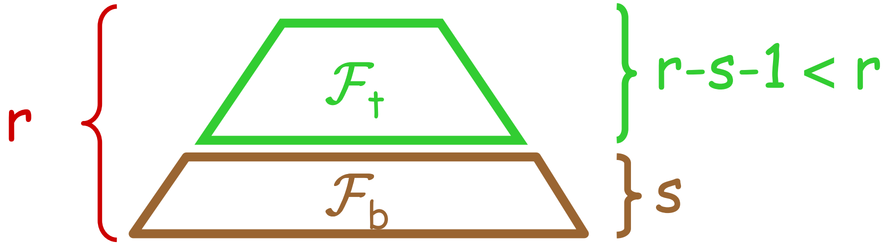


$|X_{>s}| = n_t$ $\quad$ $|\mathcal{C}_t| = m_t$

$|X_{\le s}| = n_b = n-n_t$ $\quad$ $|\mathcal{C}_b| = m_b$

$$\mathrm{cost}(\,\mathcal{C}\,) \le \mathrm{cost}(\,\mathcal{C}_t\,) + \mathrm{cost}(\,\mathcal{C}_b\,) + |X_b| - \#\mathrm{rts}(\mathcal{F}_b) + |\mathcal{C}_t|$$

$$\le f(m_t,n_t,r-s-1) + f(m_b,n_b,s) + n-n_t -$$

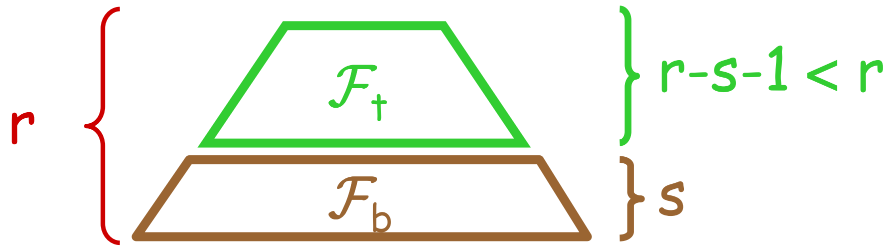


$|X_{>s}| = n_t$ $\quad |\mathcal{C}_t| = m_t$

$|X_{\leq s}| = n_b = n-n_t$ $\quad |\mathcal{C}_b| = m_b$

$$\mathrm{cost}(\,\mathcal{C}\,) \leq \mathrm{cost}(\,\mathcal{C}_t\,) + \mathrm{cost}(\,\mathcal{C}_b\,) + |X_b| - \#\mathrm{rts}(\mathcal{F}_b) + |\mathcal{C}_t|$$

$$\leq f(m_t,n_t,r-s-1) + f(m_b,n_b,s) + n-n_t - (s+1)\cdot n_t +$$



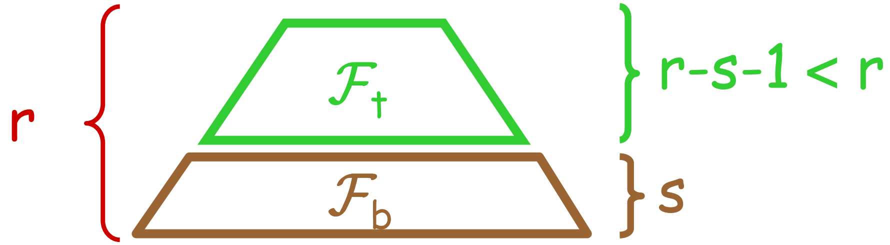
$|X_{>s}| = n_t$ $|C_t| = m_t$

$|X_{\leq s}| = n_b = n\text{-}n_t$ $|C_b| = m_b$

$$\mathrm{cost}(\,C\,) \leq \mathrm{cost}(\,C_t\,) + \mathrm{cost}(\,C_b\,) + |X_b| - \#\mathrm{rts}(\mathcal{F}_b) + |C_t|$$

$$\leq f(m_t,n_t,r\text{-}s\text{-}1) + f(m_b,n_b,s) + n\text{-}n_t - (s+1)\cdot n_t +$$

Each node in $\mathcal{F}_t$ has at least s+1 children in $\mathcal{F}_b$, and they must all be different roots of $\mathcal{F}_b$.



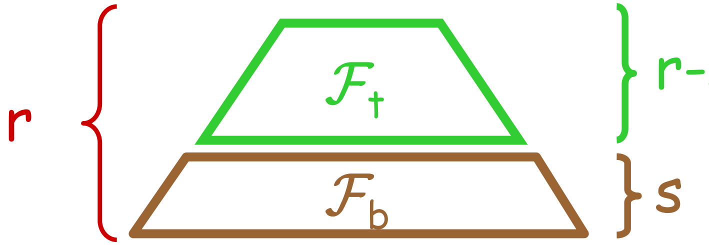
$|X_{>s}| = n_t \qquad |C_t| = m_t$

$|X_{\le s}| = n_b = n-n_t \qquad |C_b| = m_b$

$$\text{cost}(\,C\,) \le \text{cost}(\,C_t\,) + \text{cost}(\,C_b\,) + |X_b| - \#\text{rts}(\mathcal{F}_b) + |C_t|$$

$$\le f(m_t,n_t,r-s-1) + f(m_b,n_b,s) + n-n_t - (s+1)\cdot n_t + m_t$$

Each node in $\mathcal{F}_t$ has at least $s+1$ children in $\mathcal{F}_b$, and they must all be different roots of $\mathcal{F}_b$.



$|X_{>s}| = n_t$ $\quad$ $|C_t| = m_t$

$|X_{\leq s}| = n_b = n-n_t$ $\quad$ $|C_b| = m_b$

$$\mathrm{cost}(\,C\,) \leq \mathrm{cost}(\,C_t\,) + \mathrm{cost}(\,C_b\,) + |X_b| - \#\mathrm{rts}(\mathcal{F}_b) + |C_t|$$

$$\leq f(m_t,n_t,r-s-1) + f(m_b,n_b,s) + n-n_t - (s+1)\cdot n_t + m_t$$

Each node in $\mathcal{F}_t$ has at least s+1 children in $\mathcal{F}_b$, and they must all be different roots of $\mathcal{F}_b$.

$$f(m,n,r) \leq f(m_t,n_t,r-s-1) + f(m_b,n_b,s) + n - (s+2)\cdot n_t + m_t$$

$$\boxed{f(m,n,r) \leq f(m_t,n_t,r-s-1) + f(m_b,n_b,s) + n - (s+2)\cdot n_t + m_t}$$

$$n_t + n_b = n$$
$$m_t + m_b \leq m$$

$$0 \leq s < r$$

$$\boxed{f(m,n,r) \leq f(m_t,n_t,r-s-1) + f(m_b,n_b,s) + n - (s+2)\cdot n_t + m_t}$$

$$n_t + n_b = n \qquad 0 \leq s < r$$
$$m_t + m_b \leq m$$

Assume: $f(M,N,R) \leq k\cdot M + N\cdot g(R)$

$$\boxed{f(m,n,r) \;\le\; f(m_t,n_t,r-s-1) + f(m_b,n_b,s) + n - (s+2)\cdot n_t + m_t}$$

$$n_t + n_b = n \qquad 0 \le s < r$$
$$m_t + m_b \le m$$

Assume: $f(M,N,R) \le k\cdot M + N\cdot g(R)$

$$f(m,n,r) \le k\cdot m_t + n_t\cdot g(r-s-1) + f(m_b,n_b,s) + n - (s+2)\cdot n_t + m_t$$
$$\le k\cdot m_t + n_t\cdot g(r) + f(m_b,n_b,s) + n - s\cdot n_t + m_t$$

$$\boxed{f(m,n,r) \;\le\; f(m_t,n_t,r\text{-}s\text{-}1) \;+\; f(m_b,n_b,s) \;+\; n - (s+2)\cdot n_t + m_t}$$

$$n_t + n_b \;=\; n \qquad\qquad 0 \le s < r$$
$$m_t + m_b \le m$$

Assume: $f(M,N,R) \le k\cdot M + N\cdot g(R)$

$$f(m,n,r) \le k\cdot m_t + n_t\cdot g(r\text{-}s\text{-}1) + f(m_b,n_b,s) + n - (s+2)\cdot n_t + m_t$$
$$\le k\cdot m_t + n_t\cdot g(r) \;+\; f(m_b,n_b,s) + n - s\cdot n_t + m_t$$

choose $s = g(r)$

$$\boxed{f(m,n,r) \le f(m_t,n_t,r-s-1) + f(m_b,n_b,s) + n - (s+2)\cdot n_t + m_t}$$

$n_t + n_b = n$

$m_t + m_b \le m$

$0 \le s < r$

Assume: $f(M,N,R) \le k\cdot M + N\cdot g(R)$

$$f(m,n,r) \le k\cdot m_t + n_t\cdot g(r-s-1) + f(m_b,n_b,s) + n - (s+2)\cdot n_t + m_t$$

$$\le k\cdot m_t + n_t\cdot g(r) + f(m_b,n_b,s) + n - s\cdot n_t + m_t$$

choose $s = g(r)$

$$f(m,n,r) \le (k+1)\cdot m_t + f(m_b,n_b,s) + n$$

$$\le (k+1)\cdot m_t + f(m_b,n,s) + n$$

$s = g(r)$

$f(m,n,r) \leq (k+1) \cdot m_t + f(m_b,n,s) + n$

$s = g(r)$

$$f(m,n,r) \leq (k+1)\cdot m_t + f(m_b,n,s) + n \quad \Big| \quad -(k+1)\cdot(m_b+m_t)$$

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$s = g(r)$

$$f(m,n,r) \leq (k+1)\cdot m_t + f(m_b,n,s) + n \;\Big|\; -(k+1)\cdot\overbrace{(m_b+m_t)}^{m}$$

$s = g(r)$

$$f(m,n,r) \leq (k+1)\cdot m_t + f(m_b,n,s) + n \quad \Big| \quad -(k+1)\cdot\overbrace{(m_b+m_t)}^{m}$$

$$f(m,n,r) - (k+1)\cdot m \leq f(m_b,n,s) - (k+1)\cdot m_b + n$$

$s = g(r)$

$$f(m,n,r) \le (k+1)\cdot m_t + f(m_b,n,s) + n \quad \Big| \quad -(k+1)\cdot\overbrace{(m_b+m_t)}^{m}$$

$$f(m,n,r) - (k+1)\cdot m \le f(m_b,n,s) - (k+1)\cdot m_b + n$$

$$\phi(m,n,r) \le \phi(m_b,n,g(r)) + n$$

$s = g(r)$

$$f(m,n,r) \leq (k+1)\cdot m_t + f(m_b,n,s) + n \quad \Big| \quad -(k+1)\cdot\overbrace{(m_b+m_t)}^{m}$$

$$f(m,n,r) - (k+1)\cdot m \leq f(m_b,n,s) - (k+1)\cdot m_b + n$$

$$\phi(m,n,r) \leq \phi(m_b,n,g(r)) + n$$

$$\leq (\phi(m_{bb},n,g(g(r))) + n) + n$$

$s = g(r)$

$$f(m,n,r) \le (k+1)\cdot m_t + f(m_b,n,s) + n \quad \Big| \quad -(k+1)\cdot\overbrace{(m_b+m_t)}^{m}$$

$$f(m,n,r) - (k+1)\cdot m \le f(m_b,n,s) - (k+1)\cdot m_b + n$$

$$\begin{aligned} \phi(m,n,r) &\le \phi(m_b,n,g(r)) + n \\ &\le (\phi(m_{bb},n,g(g(r))) + n) + n \\ &\le ((\phi(m_{bbb},n,g(g(g(r)))) + n) + n) + n \end{aligned}$$

$$s = g(r)$$

$$f(m,n,r) \le (k+1)\cdot m_t + f(m_b,n,s) + n \quad \Bigg| \quad -(k+1)\cdot\overbrace{(m_b+m_t)}^{m}$$

$$f(m,n,r) - (k+1)\cdot m \le f(m_b,n,s) - (k+1)\cdot m_b + n$$

$$\begin{aligned}
\phi(m,n,r) &\le \phi(m_b,n,g(r)) + n \\
&\le (\phi(m_{bb},n,g(g(r))) + n) + n \\
&\le ((\phi(m_{bbb},n,g(g(g(r)))) + n) + n) + n
\end{aligned}$$

$$\phi(m,n,r) \le n \cdot g^*(r)$$

$$s = g(r)$$

$$f(m,n,r) \le (k+1)\cdot m_t + f(m_b,n,s) + n \quad \Big| \quad -(k+1)\cdot\overbrace{(m_b+m_t)}^{m}$$

$$f(m,n,r) - (k+1)\cdot m \le f(m_b,n,s) - (k+1)\cdot m_b + n$$

$$\phi(m,n,r) \le \phi(m_b,n,g(r)) + n$$

$$\le (\phi(m_{bb},n,g(g(r))) + n) + n$$

$$\le ((\phi(m_{bbb},n,g(g(g(r)))) + n) + n) + n$$

$$\phi(m,n,r) \le n \cdot g^*(r)$$

$$f(m,n,r) \le (k+1)\cdot m + n \cdot g^*(r)$$

Shifting Lemma:

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+1) \cdot m + n \cdot g^*(r)$

Shifting Lemma:

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+1) \cdot m + n \cdot g^*(r)$

Shifting Corollary:

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\cdots*}^{i}}(r)$

for any $i \geq 0$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\ldots*}^{i}}(r)$

for any $i \geq 0$

© Raimund Seidel

SAARLAND UNIVERSITY
COMPUTER SCIENCE

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\cdots*}^{i}}(r)$

for any $i \geq 0$

Trivial bound: $f(m,n,r) \leq n \cdot (r-1)$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\dots*}^{i}}(r)$

for any $i \geq 0$

Trivial bound: $f(m,n,r) \leq n \cdot (r-1)$

$= 0 \cdot m + n \cdot (r-1)$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\ldots*}^{i}}(r)$

for any $i \geq 0$

Trivial bound: $f(m,n,r) \leq n \cdot (r-1)$

$= 0 \cdot m + n \cdot (r-1)$

$g(r) = r-1$

$g^{*}(r) = r-1$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\ldots*}^{i}}(r)$

for any $i \geq 0$

Trivial bound: $f(m,n,r) \leq m + n \cdot (r-2)$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\dots*}^{i}}(r)$

for any $i \geq 0$

Trivial bound: $f(m,n,r) \leq m + n \cdot (r-2)$

$= 1 \cdot m + n \cdot (r-2)$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{** \ldots *}^{i}}(r)$

for any $i \geq 0$

Trivial bound: $f(m,n,r) \leq m + n \cdot (r-2)$

$= 1 \cdot m + n \cdot (r-2)$

$g(r) = r-2$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\cdots*}^{i}}(r)$

for any $i \geq 0$

Trivial bound: $f(m,n,r) \leq m + n \cdot (r-2)$

$= 1 \cdot m + n \cdot (r-2)$

$g(r) = r-2$

$g^*(r) = r/2$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\dots*}^{i}}(r)$

for any $i \geq 0$

Trivial bound: $f(m,n,r) \leq m + n \cdot (r-2)$

$= 1 \cdot m + n \cdot (r-2)$

$g(r) = r-2$

$g^*(r) = r/2$

$f(m,n,r) \leq 2 \cdot m + n \cdot (r/2)$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\cdots*}^{i}}(r)$

for any $i \geq 0$

Trivial bound: $f(m,n,r) \leq m + n \cdot (r-2)$

$= 1 \cdot m + n \cdot (r-2)$

$g(r) = r-2$

$g^{*}(r) = r/2$

$f(m,n,r) \leq 2 \cdot m + n \cdot (r/2)$

$g^{**}(r) = \log r$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\cdots*}^{i}}(r)$

for any $i \geq 0$

Trivial bound: $f(m,n,r) \leq m + n \cdot (r-2)$

$= 1 \cdot m + n \cdot (r-2)$

$g(r) = r-2$

$g^{*}(r) = r/2$ $\qquad f(m,n,r) \leq 2 \cdot m + n \cdot (r/2)$

$g^{**}(r) = \log r$ $\qquad f(m,n,r) \leq 3 \cdot m + n \cdot \log r$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\cdots*}^{i}}(r)$

for any $i \geq 0$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\dots*}^{i}}(r)$

for any $i \geq 0$

We know bound: $f(m,n,r) \leq 3 \cdot m + n \cdot \log r$

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\cdots*}^{i}}(r)$

for any $i \geq 0$

We know bound: $f(m,n,r) \leq 3 \cdot m + n \cdot \log r$

Therefore for any $i \geq 0$:

$$f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{**\cdots*}^{i}}(r)$$

For any $i \geq 0$: $f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{**\dots*}^{i}}(r)$

For any $i \geq 0$: $f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{**\cdots*}^{i}}(r)$

Choice of i :

For any $i \geq 0$: $f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{** \cdots *}^{i}}(r)$

Choice of i :

Define $\alpha(r) = \min\{\, i \mid \log^{\overbrace{** \cdots *}^{i}}(r) \leq i \,\}$

For any $i \geq 0$: $f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{**\dots*}^{i}}(r)$

Choice of i :

Define $\alpha(r) = \min\{ i \mid \log^{\overbrace{**\dots*}^{i}}(r) \leq i \}$

Here is your definition of the Inverse Ackermann Function !!

For any $i \geq 0$: $f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{**\ldots*}^{i}}(r)$

Choice of i :

Define $\alpha(r) = \min\{ i \mid \log^{\overbrace{**\ldots*}^{i}}(r) \leq i \}$

$$f(m,n,r) \leq (m+n)(3+\alpha(r))$$

For any $i \geq 0$: $f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{**\dots*}^{i}}(r)$

Choice of i :

Define $\alpha(r) = \min\{ i \mid \log^{\overbrace{**\dots*}^{i}}(r) \leq i \}$

$$f(m,n,r) \leq (m+n)(3+\alpha(r))$$

$$\leq (m+n)(3+\alpha(\log n))$$

For any $i \geq 0$: $f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{**\cdots*}^{i}}(r)$

Choice of i :

For any $i \geq 0$: $f(m,n,r) \leq (3+i)\cdot m + n\cdot \log^{\overbrace{**\cdots*}^{i}}(r)$

Choice of i :

For $t \geq 1$ define $\alpha_t(r) = \min\{ i \mid \log^{\overbrace{**\cdots*}^{i}}(r) \leq t \}$

For any $i \geq 0$: $f(m,n,r) \leq (3+i)\cdot m + n\cdot \log^{\overbrace{**\cdots*}^{i}}(r)$

Choice of i :

For $t \geq 1$ define $\alpha_t(r) = \min\{\, i \mid \log^{\overbrace{**\cdots*}^{i}}(r) \leq t \,\}$

Here is a parametrized definition of the Inverse Ackermann Function !!

$$\boxed{\text{For any } i \geq 0 : \; f(m,n,r) \leq (3+i)\cdot m + n\cdot \log^{\overbrace{**\cdots*}^{i}}(r)}$$

Choice of i :

For $t \geq 1$ define $\alpha_t(r) = \min\{\, i \mid \log^{\overbrace{**\cdots*}^{i}}(r) \leq t \,\}$

$$f(m,n,r) \leq (3+\alpha_t(r))\cdot m + n\cdot t$$

$$\boxed{\text{For any } i \geq 0 : \quad f(m,n,r) \leq (3+i)\cdot m + n\cdot \log^{\overbrace{**\cdots*}^{i}}(r)}$$

Choice of i :

For $t \geq 1$ define $\alpha_t(r) = \min\{\, i \mid \log^{\overbrace{**\cdots*}^{i}}(r) \leq t \,\}$

$$f(m,n,r) \leq (3+\alpha_t(r))\cdot m + n\cdot t$$

choose $t = 1+m/n$

For any $i \geq 0$: $f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{** \dots *}^{i}}(r)$

Choice of i :

For $t \geq 1$ define $\alpha_t(r) = \min\{\, i \mid \log^{\overbrace{** \dots *}^{i}}(r) \leq t \,\}$

$$f(m,n,r) \leq (3+\alpha_t(r)) \cdot m + n \cdot t$$

choose $t = 1+m/n$

$$f(m,n,r) \leq (4+\alpha_{1+m/n}(r)) \cdot m + n$$

For any $i \geq 0$: $f(m,n,r) \leq (3+i)\cdot m + n\cdot \log^{\overbrace{**\ldots*}^{i}}(r)$

Choice of i :

For $t \geq 1$ define $\alpha_t(r) = \min\{\, i \mid \log^{\overbrace{**\ldots*}^{i}}(r) \leq t \,\}$

$$f(m,n,r) \leq (3+\alpha_t(r))\cdot m + n\cdot t$$

choose $t = 1+m/n$

$$f(m,n,r) \leq (4+\alpha_{1+m/n}(r))\cdot m + n$$

$$\leq (4+\alpha_{1+m/n}(\log n))\cdot m + n$$

Bob Tarjan 1975

Theorem:

Any sequence of m Union, Find operations in a universe of n elements that uses linking by rank and path compression takes time at most

$$O(\ m\cdot\alpha(m,n) + n\ )$$

Bob Tarjan 1975

Theorem:

Any sequence of m Union, Find operations in a universe of n elements that uses linking by rank and path compression takes time at most

$$O(\ m\cdot\alpha(m,n) + n\ )$$

$$\boxed{f(m,n,r) \le (4+\alpha_{1+m/n}(\log n))\cdot m + n}$$

Bob Tarjan 1975

Theorem:

Any sequence of m Union, Find operations in a universe of n elements that uses linking by rank and path compression takes time at most

$$O(\ m \cdot \alpha(m,n) + n\ )$$

$$\boxed{f(m,n,r) \leq (4+\alpha_{1+m/n}(\log n)) \cdot m + n}$$

$$\alpha(m,n) = \alpha_{1+m/n}(\log n)$$

Shifting Lemma:

What to remember:

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+1) \cdot m + n \cdot g^{*}(r)$

Shifting Corollary:

If $f(m,n,r) \leq k \cdot m + n \cdot g(r)$

then also $f(m,n,r) \leq (k+i) \cdot m + n \cdot g^{\overbrace{**\dots*}^{i}}(r)$

for any $i \geq 0$

Definition of α:

$$\alpha(r) = \min\{\, i \mid \log^{\overbrace{**\dots*}^{i}}(r) \leq i \,\}$$

Odds and Ends

Odds and Ends

We used $f(m,n,r) \leq 1 \cdot m + n \cdot (r-2)$

Odds and Ends

We used $f(m,n,r) \leq 1 \cdot m + n \cdot (r-2)$ to get

$$\boxed{\text{for any } i \geq 0 : \quad f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{**\cdots*}^{i}}(r)}$$

Odds and Ends

We used $f(m,n,r) \leq 1 \cdot m + n \cdot (r-2)$ to get

$$\boxed{\text{for any } i \geq 0 : \ f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{**\cdots*}^{i}}(r)}$$

Actually $f(m,n,r) \leq 1 \cdot m + n \cdot \log r$

Odds and Ends

We used $f(m,n,r) \leq 1 \cdot m + n \cdot (r-2)$ to get

for any $i \geq 0$: $f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{**\dots*}^{i}}(r)$

Actually $f(m,n,r) \leq 1 \cdot m + n \cdot \log r$ (Exercise)

Odds and Ends

We used $f(m,n,r) \leq 1 \cdot m + n \cdot (r-2)$ to get

$$\boxed{\text{for any } i \geq 0 : \; f(m,n,r) \leq (3+i) \cdot m + n \cdot \log^{\overbrace{**\cdots*}^{i}}(r)}$$

Actually $f(m,n,r) \leq 1 \cdot m + n \cdot \log r$ (Exercise)

and therefore

$$\boxed{\text{for any } i \geq 0 : \; f(m,n,r) \leq (1+i) \cdot m + n \cdot \log^{\overbrace{**\cdots*}^{i}}(r)}$$

Odds and Ends

Actually $f(m,n,r) \leq 1 \cdot m + n \cdot \log^* r$ (difficult Exercise)

and therefore

For any $i \geq 0$: $f(m,n,r) \leq i \cdot m + n \cdot \log^{\overbrace{**\dots*}^{i}}(r)$

Odds and Ends

$f(m,n,r)$ for small values of r

Odds and Ends

$f(m,n,r)$ for small values of r

$f(m,n,0) = 0$ $f(m,n,1) = 0$ $f(m,n,2) \leq m$

Odds and Ends

$f(m,n,r)$ for small values of r

$f(m,n,0) = 0$ $f(m,n,1) = 0$ $f(m,n,2) \leq m$

$f(m,n,r) \leq m + n$ for $r \leq 8$, i.e. for $n < 512$

Odds and Ends

$f(m,n,r)$ for small values of r

$f(m,n,0) = 0$ $f(m,n,1) = 0$ $f(m,n,2) \leq m$

$f(m,n,r) \leq m + n$ for $r \leq 8$, i.e. for $n<512$

$f(m,n,r) \leq m + 2n$ for $r \leq 202$, i.e. for $n<2^{203}$

Odds and Ends

$f(m,n,r)$ for small values of r

$f(m,n,0) = 0$ $f(m,n,1) = 0$ $f(m,n,2) \leq m$

$\boxed{f(m,n,r) \leq m + n \quad \text{for } r \leq 8, \text{ i.e. for } n<512}$

$\boxed{f(m,n,r) \leq m + 2n \quad \text{for } r \leq 202, \text{ i.e. for } n<2^{203}}$

(difficult exercises)

Odds and Ends

Similar proof for $O(m \cdot \alpha(m,n) + n)$ bound also works for

- * linking by weight and path compression
- * linking by rank and generalized path compaction

Odds and Ends

Similar proof for $O(m \cdot \alpha(m,n) + n)$ bound also works for

* linking by weight and path compression

* linking by rank and generalized path compaction

Open problem:

simple top-down approach for proving **lower bounds**