Automata, languages and monoids II.

Adéla Haníková

November 20, 2014

Definitions

Definition 1

For each integer $n \geq 0$, we define an equivalence relation $\sim_n$ on A^* by $u \sim_n v$ if and only if u and v have the same set of subwords of length less than or equal to n.

Definitions

Definition 1

For each integer $n \geq 0$, we define an equivalence relation $\sim_n$ on A^* by $u \sim_n v$ if and only if u and v have the same set of subwords of length less than or equal to n.

- For example $abbac \sim_1 abc$ or $ababab \sim_3 bababa$

Definitions

Definition 1

For each integer $n \geq 0$, we define an equivalence relation $\sim_n$ on A^* by $u \sim_n v$ if and only if u and v have the same set of subwords of length less than or equal to n.

- For example $abbac \sim_1 abc$ or $ababab \sim_3 bababa$
- $\sim_n$ is a congruence.

Definitions

Definition 1

For each integer $n \geq 0$, we define an equivalence relation $\sim_n$ on A^* by $u \sim_n v$ if and only if u and v have the same set of subwords of length less than or equal to n.

- For example $abbac \sim_1 abc$ or $ababab \sim_3 bababa$
- $\sim_n$ is a congruence.

Definition 2

A language is called *piecewise testable* if it is the union of classes modulo $\sim_n$ for a certain integer n.

Proposition 3

A language L of A^* is piecewise testable if and only if it is in the boolean algebra generated by the languages of the form $A^*a_1A^*a_2\dots A^*a_mA^*$ where $0 \leq m$ and the a_i are letters.

Proposition 3

A language L of A^* is piecewise testable if and only if it is in the boolean algebra generated by the languages of the form $A^*a_1A^*a_2 \dots A^*a_mA^*$ where $0 \leq m$ and the a_i are letters.

Piecewise testable languages are regular.

Proof.

- "$\Rightarrow$": $\{v \in A^* | v \sim_n u\} = (\cap A^* a_1 A^* a_2 \ldots a_m A^*) \setminus (\cup A^* a_1 A^* a_2 \ldots a_m A^*)$

$\square$

Proof.

- "$\Rightarrow$": $\{v \in A^* | v \sim_n u\} = (\cap A^* a_1 A^* a_2 \dots a_m A^*) \setminus (\cup A^* a_1 A^* a_2 \dots a_m A^*)$
- intersection over the set of m-tuples $(a_1, \dots, a_m)$ such that $0 \leq m \leq n$ and $a_1, \dots, a_m$ is a subword of u

$\square$

Proof.

- "$\Rightarrow$": $\{v \in A^* | v \sim_n u\} = (\cap A^* a_1 A^* a_2 \dots a_m A^*) \setminus (\cup A^* a_1 A^* a_2 \dots a_m A^*)$
- intersection over the set of m-tuples $(a_1, \dots, a_m)$ such that $0 \leq m \leq n$ and $a_1, \dots, a_m$ is a subword of u
- union over the set of m-tuples $(a_1, \dots, a_m)$ such that $0 \leq m \leq n$ and $a_1, \dots, a_m$ is not a subword of u

$\square$

Proof.

- $" \Rightarrow ":$ $\{v \in A^* | v \sim_n u\} = (\bigcap A^* a_1 A^* a_2 \dots a_m A^*) \setminus (\bigcup A^* a_1 A^* a_2 \dots a_m A^*)$
- intersection over the set of m-tuples $(a_1, \dots, a_m)$ such that $0 \le m \le n$ and $a_1, \dots, a_m$ is a subword of u
- union over the set of m-tuples $(a_1, \dots, a_m)$ such that $0 \le m \le n$ and $a_1, \dots, a_m$ is not a subword of u
- $" \Leftarrow "$ let $L = A^* a_1 A^* a_2 \dots a_n A^*$ and $u \in L$

$\square$

Proof.

- "$\Rightarrow$": $\{v \in A^* | v \sim_n u\} = (\cap A^* a_1 A^* a_2 \dots a_m A^*) \setminus (\cup A^* a_1 A^* a_2 \dots a_m A^*)$
- intersection over the set of m-tuples $(a_1, \dots, a_m)$ such that $0 \leq m \leq n$ and $a_1, \dots, a_m$ is a subword of u
- union over the set of m-tuples $(a_1, \dots, a_m)$ such that $0 \leq m \leq n$ and $a_1, \dots, a_m$ is not a subword of u
- "$\Leftarrow$ " let $L = A^* a_1 A^* a_2 \dots a_n A^*$ and $u \in L$
- if $u \sim_n v$, $a_1, \dots, a_n$ is a subword of v and $v \in L$

$\square$

Proof.

- "$\Rightarrow$": $\{v \in A^* | v \sim_n u\} = (\cap A^* a_1 A^* a_2 \dots a_m A^*) \setminus (\cup A^* a_1 A^* a_2 \dots a_m A^*)$
- intersection over the set of m-tuples $(a_1, \dots, a_m)$ such that $0 \leq m \leq n$ and $a_1, \dots, a_m$ is a subword of u
- union over the set of m-tuples $(a_1, \dots, a_m)$ such that $0 \leq m \leq n$ and $a_1, \dots, a_m$ is not a subword of u
- "$\Leftarrow$" let $L = A^* a_1 A^* a_2 \dots a_n A^*$ and $u \in L$
- if $u \sim_n v$, $a_1, \dots, a_n$ is a subword of v and $v \in L$
- so L is a finite union of classes modulo $\sim_n$

$\square$

Proposition 4

Let $u, v \in A^*$ and $a \in A$. Then $uav \sim_{2n-1} uv$ implies $ua \sim_n u$ or $av \sim_n v$.

Proposition 4

Let $u, v \in A^*$ and $a \in A$. Then $uav \sim_{2n-1} uv$ implies $ua \sim_n u$ or $av \sim_n v$.

Proof.

By contradiction

- suppose $ua \nsim_n u$ and $av \nsim_n v$.

□

Proposition 4

Let $u, v \in A^*$ and $a \in A$. Then $uav \sim_{2n-1} uv$ implies $ua \sim_n u$ or $av \sim_n v$.

Proof.

By contradiction

- suppose $ua \nsim_n u$ and $av \nsim_n v$.
- $x = x'a$ subword of ua and $y = ay'$ subword of av, length $\leq n$

□

Proposition 4

Let $u, v \in A^*$ and $a \in A$. Then $uav \sim_{2n-1} uv$ implies $ua \sim_n u$ or $av \sim_n v$.

Proof.

By contradiction

- suppose $ua \nsim_n u$ and $av \nsim_n v$.
- $x = x'a$ subword of ua and $y = ay'$ subword of av, length $\leq n$
- $x'ay'$ has length $\leq 2n - 1$ and is a subword of uav and not uv

$\square$

Definition 5

If u is a word, we denote by $alph(u)$ the alphabet of u, i.e. the set of letters a such that $|u|_a > 0$.

Definition 5

If u is a word, we denote by $alph(u)$ the alphabet of u, i.e. the set of letters a such that $|u|_a > 0$.

Proposition 6

Let $u, v \in A^*$ and $n > 0$. Then $u \sim_n vu$ if and only if there there exist $u_1, \ldots, u_n \in A^*$ such that $u = u_1 \ldots u_n$ and $alph(v) \subseteq alph(u_1) \subseteq \cdots \subseteq alph(u_n)$.

Definition 5

If u is a word, we denote by $alph(u)$ the alphabet of u, i.e. the set of letters a such that $|u|_a > 0$.

Proposition 6

Let $u, v \in A^*$ and $n > 0$. Then $u \sim_n vu$ if and only if there there exist $u_1, \ldots, u_n \in A^*$ such that $u = u_1 \ldots u_n$ and $alph(v) \subseteq alph(u_1) \subseteq \cdots \subseteq alph(u_n)$.

Proof

- the result is trivial, if $u = \lambda \Rightarrow$ suppose $u \in A^+$

Definition 5

If u is a word, we denote by $alph(u)$ the alphabet of u, i.e. the set of letters a such that $|u|_a > 0$.

Proposition 6

Let $u, v \in A^*$ and $n > 0$. Then $u \sim_n vu$ if and only if there there exist $u_1, \ldots, u_n \in A^*$ such that $u = u_1 \ldots u_n$ and $alph(v) \subseteq alph(u_1) \subseteq \cdots \subseteq alph(u_n)$.

Proof

- the result is trivial, if $u = \lambda \Rightarrow$ suppose $u \in A^+$
- necessarility $(\Rightarrow)$ and sufficiency$(\Leftarrow)$, both by induction on n

Proof: Necessarility ($\Rightarrow$)

- $n = 1$, then
  $u \sim_1 vu \Rightarrow alph(u) = alph(vu) \Rightarrow alph(v) \subseteq alph(u)$

Proof: Necessarility ($\Rightarrow$)

- $n = 1$, then
  $u \sim_1 vu \Rightarrow alph(u) = alph(vu) \Rightarrow alph(v) \subseteq alph(u)$
- suppose $u \sim_{n+1} vu$ and let u_{n+1} be shortest suffix such that $alph(u_{n+1}) = alph(u)$

Proof: Necessarility ($\Rightarrow$)

- $n = 1$, then
  $u \sim_1 vu \Rightarrow alph(u) = alph(vu) \Rightarrow alph(v) \subseteq alph(u)$
- suppose $u \sim_{n+1} vu$ and let u_{n+1} be shortest suffix such that $alph(u_{n+1}) = alph(u)$
- $u_{n+1} = au', a \in A$ so $a \notin alph(u')$ and $u = wau'$

Proof: Necessarility ($\Rightarrow$)

- $n = 1$, then
  $u \sim_1 vu \Rightarrow alph(u) = alph(vu) \Rightarrow alph(v) \subseteq alph(u)$
- suppose $u \sim_{n+1} vu$ and let u_{n+1} be shortest suffix such that $alph(u_{n+1}) = alph(u)$
- $u_{n+1} = au', a \in A$ so $a \notin alph(u')$ and $u = wau'$
- we want to show $w \sim_n vw$

Proof: Necessarility ($\Rightarrow$)

- $n = 1$, then
  $u \sim_1 vu \Rightarrow alph(u) = alph(vu) \Rightarrow alph(v) \subseteq alph(u)$
- suppose $u \sim_{n+1} vu$ and let u_{n+1} be shortest suffix such that $alph(u_{n+1}) = alph(u)$
- $u_{n+1} = au', a \in A$ so $a \notin alph(u')$ and $u = wau'$
- we want to show $w \sim_n vw$
- then from the inductive hypothesis $\exists u_1, \ldots, u_n \in A^*$ such that $u_1 \ldots u_n = w$ and
  $alph(v) \subseteq alph(u_1) \subseteq \cdots \subseteq alph(u_n)$

- $n = 1$, then
  $u \sim_1 vu \Rightarrow alph(u) = alph(vu) \Rightarrow alph(v) \subseteq alph(u)$
- suppose $u \sim_{n+1} vu$ and let u_{n+1} be shortest suffix such that $alph(u_{n+1}) = alph(u)$
- $u_{n+1} = au', a \in A$ so $a \notin alph(u')$ and $u = wau'$
- we want to show $w \sim_n vw$
- then from the inductive hypothesis $\exists u_1, \ldots, u_n \in A^*$ such that $u_1 \ldots u_n = w$ and
  $alph(v) \subseteq alph(u_1) \subseteq \cdots \subseteq alph(u_n)$
- $u = wu_{n+1}$ and $alph(u_n) \subseteq alph(u) = alph(u_{n+1})$

Proof: Necessarility (⇒)

- $u \sim_{n+1} vu,\ u = wau' \Rightarrow w \sim_n vw$

Proof: Necessarility ($\Rightarrow$)

- $u \sim_{n+1} vu,\ u = wau' \Rightarrow w \sim_n vw$
- trivially every subword of w is subword of vw

Proof: Necessarility ($\Rightarrow$)

- $u \sim_{n+1} vu$, $u = wau' \Rightarrow w \sim_n vw$
- trivially every subword of w is subword of vw
- set x a subword of vw, length of $x \leq n$

Proof: Necessarility ($\Rightarrow$)

- $u \sim_{n+1} vu$, $u = wau' \Rightarrow w \sim_n vw$
- trivially every subword of w is subword of vw
- set x a subword of vw, length of $x \leq n$
- xa subword of vu and therefore u and therefore wa

Proof: Sufficiency ($\Leftarrow$)

- if $n = 1$ then $u_1 = u$ and
  $alph(v) \subseteq alph(u) \Rightarrow alph(u) = alph(vu)$ i.e. $u \sim_1 vu$

Proof: Sufficiency ($\Leftarrow$)

- if $n = 1$ then $u_1 = u$ and $alph(v) \subseteq alph(u) \Rightarrow alph(u) = alph(vu)$ i.e. $u \sim_1 vu$
- suppose $u = u_1 \dots u_{n+1}$ and $alph(v) \subseteq alph(u_1) \subseteq \cdots \subseteq alph(u_{n+1})$

Proof: Sufficiency ($\Leftarrow$)

- if $n = 1$ then $u_1 = u$ and $alph(v) \subseteq alph(u) \Rightarrow alph(u) = alph(vu)$ i.e. $u \sim_1 vu$
- suppose $u = u_1 \dots u_{n+1}$ and $alph(v) \subseteq alph(u_1) \subseteq \dots \subseteq alph(u_{n+1})$
- then $alph(vu) = alph(u) = alph(u_{n+1})$ and $u_1 \dots u_n \sim_n vu_1 \dots u_n$ from the inductive hypothesis

Proof: Sufficiency ($\Leftarrow$)

- if $n = 1$ then $u_1 = u$ and
  $alph(v) \subseteq alph(u) \Rightarrow alph(u) = alph(vu)$ i.e. $u \sim_1 vu$
- suppose $u = u_1 \dots u_{n+1}$ and
  $alph(v) \subseteq alph(u_1) \subseteq \cdots \subseteq alph(u_{n+1})$
- then $alph(vu) = alph(u) = alph(u_{n+1})$ and
  $u_1 \dots u_n \sim_n vu_1 \dots u_n$ from the inductive hypothesis
- let x be a non-empty subword of vu of length $\leq n + 1$

Proof: Sufficiency ($\Leftarrow$)

- if $n = 1$ then $u_1 = u$ and $alph(v) \subseteq alph(u) \Rightarrow alph(u) = alph(vu)$ i.e. $u \sim_1 vu$
- suppose $u = u_1 \dots u_{n+1}$ and $alph(v) \subseteq alph(u_1) \subseteq \cdots \subseteq alph(u_{n+1})$
- then $alph(vu) = alph(u) = alph(u_{n+1})$ and $u_1 \dots u_n \sim_n vu_1 \dots u_n$ from the inductive hypothesis
- let x be a non-empty subword of vu of length $\leq n + 1$
- set x' the longest suffix of x such that x' is a subword of u_{n+1} and put $x = x''x'$

- if $n = 1$ then $u_1 = u$ and $alph(v) \subseteq alph(u) \Rightarrow alph(u) = alph(vu)$ i.e. $u \sim_1 vu$
- suppose $u = u_1 \dots u_{n+1}$ and $alph(v) \subseteq alph(u_1) \subseteq \dots \subseteq alph(u_{n+1})$
- then $alph(vu) = alph(u) = alph(u_{n+1})$ and $u_1 \dots u_n \sim_n vu_1 \dots u_n$ from the inductive hypothesis
- let x be a non-empty subword of vu of length $\leq n+1$
- set x' the longest suffix of x such that x' is a subword of u_{n+1} and put $x = x''x'$
- x' is nonempty and consequently x'' is a subword of length $\leq n$ of $vu_1 \dots u_n$

- if $n = 1$ then $u_1 = u$ and $alph(v) \subseteq alph(u) \Rightarrow alph(u) = alph(vu)$ i.e. $u \sim_1 vu$
- suppose $u = u_1 \dots u_{n+1}$ and $alph(v) \subseteq alph(u_1) \subseteq \cdots \subseteq alph(u_{n+1})$
- then $alph(vu) = alph(u) = alph(u_{n+1})$ and $u_1 \dots u_n \sim_n vu_1 \dots u_n$ from the inductive hypothesis
- let x be a non-empty subword of vu of length $\leq n+1$
- set x' the longest suffix of x such that x' is a subword of u_{n+1} and put $x = x''x'$
- x' is nonempty and consequently x'' is a subword of length $\leq n$ of $vu_1 \dots u_n$
- so x'' is a subword of u

Proof: Sufficiency ($\Leftarrow$)

- if $n = 1$ then $u_1 = u$ and $alph(v) \subseteq alph(u) \Rightarrow alph(u) = alph(vu)$ i.e. $u \sim_1 vu$
- suppose $u = u_1 \dots u_{n+1}$ and $alph(v) \subseteq alph(u_1) \subseteq \cdots \subseteq alph(u_{n+1})$
- then $alph(vu) = alph(u) = alph(u_{n+1})$ and $u_1 \dots u_n \sim_n vu_1 \dots u_n$ from the inductive hypothesis
- let x be a non-empty subword of vu of length $\leq n+1$
- set x' the longest suffix of x such that x' is a subword of u_{n+1} and put $x = x''x'$
- x' is nonempty and consequently x'' is a subword of length $\leq n$ of $vu_1 \dots u_n$
- so x'' is a subword of u
- therefore $u \sim_{n+1} vu$

Corollary

For every $u, v \in A^*$, we have $(uv)^n u \sim_n (uv)^n \sim_n v(uv)^n$.

Proposition 7

If $f \sim_n g$, there exists a word h such that f is a subword of h, g is a subword of h and $f \sim_n h \sim_n g$.

Proposition 7

If $f \sim_n g$, there exists a word h such that f is a subword of h, g is a subword of h and $f \sim_n h \sim_n g$.

Proof

The proof is achieved by induction on $k = |f| + |g| - 2|f \wedge g|$ where $f \wedge g$ is the largest prefix common to f and g.

- if $k = 0$, then $f = g$ and so $f = h = g$

Proposition 7

If $f \sim_n g$, there exists a word h such that f is a subword of h, g is a subword of h and $f \sim_n h \sim_n g$.

Proof

The proof is achieved by induction on $k = |f| + |g| - 2|f \wedge g|$ where $f \wedge g$ is the largest prefix common to f and g.

- if $k = 0$, then $f = g$ and so $f = h = g$
- if g is a subword of f or vice versa then the result is also obvious

- $f = uav, g = ubw, a, b \in A, a \neq b$

- $f = uav, g = ubw, a, b \in A, a \neq b$
- we shall show that $ubw \sim_n ubav$ or $uav \sim_n uabw$

- $f = uav, g = ubw, a, b \in A, a \neq b$
- we shall show that $ubw \sim_n ubav$ or $uav \sim_n uabw$
- suppose that $f = uav \sim_n uabw$ then

- $f = uav, g = ubw, a, b \in A, a \neq b$
- we shall show that $ubw \sim_n ubav$ or $uav \sim_n uabw$
- suppose that $f = uav \sim_n uabw$ then
- $|uav| + |uabw| - 2|uav \wedge uabw| \leq |f| + |g| + 1 - 2|ua| \leq |f| + |g| + 1 - (2|f \wedge g| + 2) < k$

- $f = uav, g = ubw, a, b \in A, a \neq b$
- we shall show that $ubw \sim_n ubav$ or $uav \sim_n uabw$
- suppose that $f = uav \sim_n uabw$ then
- $|uav| + |uabw| - 2|uav \wedge uabw| \leq |f| + |g| + 1 - 2|ua| \leq |f| + |g| + 1 - (2|f \wedge g| + 2) < k$
- from inductive hypothesis $\exists h$ such that $f \sim_n h \sim_n uabw$

- $f = uav, g = ubw, a, b \in A, a \neq b$
- we shall show that $ubw \sim_n ubav$ or $uav \sim_n uabw$
- suppose that $f = uav \sim_n uabw$ then
- $|uav| + |uabw| - 2|uav \wedge uabw| \leq |f| + |g| + 1 - 2|ua| \leq |f| + |g| + 1 - (2|f \wedge g| + 2) < k$
- from inductive hypothesis $\exists h$ such that $f \sim_n h \sim_n uabw$
- the proposition follows from this, since g is a subword of $uabw$

- prove that $ubw \sim_n ubav$ or $uav \sim_n uabw$

- prove that $ubw \sim_n ubav$ or $uav \sim_n uabw$
- by contradiction: suppose none of these assertions is true

- prove that $ubw \sim_n ubav$ or $uav \sim_n uabw$
- by contradiction: suppose none of these assertions is true
- then there exist a word r of length $\leq n$ which is a subword of $ubav$ but not ubw

- prove that $ubw \sim_n ubav$ or $uav \sim_n uabw$
- by contradiction: suppose none of these assertions is true
- then there exist a word r of length $\leq n$ which is a subword of $ubav$ but not ubw
- and a word s of length $\leq n$ which is a subword of $uabw$ but not uav

- prove that $ubw \sim_n ubav$ or $uav \sim_n uabw$
- by contradiction: suppose none of these assertions is true
- then there exist a word r of length $\leq n$ which is a subword of $ubav$ but not ubw
- and a word s of length $\leq n$ which is a subword of $uabw$ but not uav
- $r = r_1 b r_2$ where r_1 is a subword of u and r_2 of av

- prove that $ubw \sim_n ubav$ or $uav \sim_n uabw$
- by contradiction: suppose none of these assertions is true
- then there exist a word r of length $\leq n$ which is a subword of $ubav$ but not ubw
- and a word s of length $\leq n$ which is a subword of $uabw$ but not uav
- $r = r_1 b r_2$ where r_1 is a subword of u and r_2 of av
- $s = s_1 a s_2$ where s_1 is a subword of u and s_2 of bw

- $r_1 b$ is not a subword of u and $s_1 a$ is not a subword of u

- $r_1 b$ is not a subword of u and $s_1 a$ is not a subword of u
- r_2 subword of $av \Rightarrow r_2 = r_2'' r_2'$ where $r_2'' = \lambda$ or a and r_2' subword of v

- $r_1 b$ is not a subword of u and $s_1 a$ is not a subword of u
- r_2 subword of $av \Rightarrow r_2 = r_2'' r_2'$ where $r_2'' = \lambda$ or a and r_2' subword of v
- likewise $s_2 = s_2'' s_2'$, $s_2'' = \lambda$ or b, s_2' subword of u

- $r_1 b$ is not a subword of u and $s_1 a$ is not a subword of u
- r_2 subword of $av \Rightarrow r_2 = r_2'' r_2'$ where $r_2'' = \lambda$ or a and r_2' subword of v
- likewise $s_2 = s_2'' s_2'$, $s_2'' = \lambda$ or b, s_2' subword of u
- $|r_1 b s_2'| + |s_1 a r_2'| \leq |r_1 a s_2| + |s_1 b r_2| \leq |r| + |s| \leq 2n$

- $r_1 b$ is not a subword of u and $s_1 a$ is not a subword of u
- r_2 subword of $av \Rightarrow r_2 = r_2'' r_2'$ where $r_2'' = \lambda$ or a and r_2' subword of v
- likewise $s_2 = s_2'' s_2'$, $s_2'' = \lambda$ or b, s_2' subword of u
- $|r_1 b s_2'| + |s_1 a r_2'| \leq |r_1 a s_2| + |s_1 b r_2| \leq |r| + |s| \leq 2n$
- so one of the words has length $\leq n$

- $r_1 b$ is not a subword of u and $s_1 a$ is not a subword of u
- r_2 subword of $av \Rightarrow r_2 = r_2'' r_2'$ where $r_2'' = \lambda$ or a and r_2' subword of v
- likewise $s_2 = s_2'' s_2'$, $s_2'' = \lambda$ or b, s_2' subword of u
- $|r_1 b s_2'| + |s_1 a r_2'| \leq |r_1 a s_2| + |s_1 b r_2| \leq |r| + |s| \leq 2n$
- so one of the words has length $\leq n$
- let it be $r_1 b s_2'$, it is a subword of $ubw = g$ and $f = uav$

- $r_1 b$ is not a subword of u and $s_1 a$ is not a subword of u
- r_2 subword of $av \Rightarrow r_2 = r_2'' r_2'$ where $r_2'' = \lambda$ or a and r_2' subword of v
- likewise $s_2 = s_2'' s_2'$, $s_2'' = \lambda$ or b, s_2' subword of u
- $|r_1 bs_2'| + |s_1 ar_2'| \leq |r_1 as_2| + |s_1 br_2| \leq |r| + |s| \leq 2n$
- so one of the words has length $\leq n$
- let it be $r_1 bs_2'$, it is a subword of $ubw = g$ and $f = uav$
- but $r_1 b$ is not a subword of $u \Rightarrow bs_2'$ subword of v

- $r_1 b$ is not a subword of u and $s_1 a$ is not a subword of u
- r_2 subword of $av \Rightarrow r_2 = r_2'' r_2'$ where $r_2'' = \lambda$ or a and r_2' subword of v
- likewise $s_2 = s_2'' s_2'$, $s_2'' = \lambda$ or b, s_2' subword of u
- $|r_1 b s_2'| + |s_1 a r_2'| \leq |r_1 a s_2| + |s_1 b r_2| \leq |r| + |s| \leq 2n$
- so one of the words has length $\leq n$
- let it be $r_1 b s_2'$, it is a subword of $ubw = g$ and $f = uav$
- but $r_1 b$ is not a subword of $u \Rightarrow bs_2'$ subword of v
- thus $s = s_1 a s_2$ subword of $uav = f$ — a contradiction

- let $f = a^3b^3a^3b^3$ and $g = a^2b^4a^4b^2$, $f \sim_4 g$

- let $f = a^3b^3a^3b^3$ and $g = a^2b^4a^4b^2$, $f \sim_4 g$
- we will find h applying the algorithm from the proof

- let $f = a^3b^3a^3b^3$ and $g = a^2b^4a^4b^2$, $f \sim_4 g$
- we will find h applying the algorithm from the proof
- $f = \overbrace{(aa)}^{u}\,a\,\overbrace{(b^3a^3b^3)}^{v} \sim_4 (aa)b\,\overbrace{(b^3a^4b^2)}^{w} = g$ so

- let $f = a^3b^3a^3b^3$ and $g = a^2b^4a^4b^2$, $f \sim_4 g$
- we will find h applying the algorithm from the proof
- $f = \overbrace{(aa)}^{u}\, a\, \overbrace{(b^3a^3b^3)}^{v} \sim_4 (aa)b\,\overbrace{(b^3a^4b^2)}^{w} = g$ so
- $(aa)a(b^3a^3b^3) \sim_4 (aa)ab(b^3a^4b^2)$ or $(aa)b(b^3a^4b^2) \sim_4 (aa)ba(b^3a^3b^3)$

- let $f = a^3b^3a^3b^3$ and $g = a^2b^4a^4b^2$, $f \sim_4 g$
- we will find h applying the algorithm from the proof
- $f = \overbrace{(aa)}^{u}\, a\, \overbrace{(b^3a^3b^3)}^{v} \sim_4 (aa)b\, \overbrace{(b^3a^4b^2)}^{w} = g$ so
- $(aa)a(b^3a^3b^3) \sim_4 (aa)ab(b^3a^4b^2)$ or $(aa)b(b^3a^4b^2) \sim_4 (aa)ba(b^3a^3b^3)$

- let $f = a^3b^3a^3b^3$ and $g = a^2b^4a^4b^2$, $f \sim_4 g$
- we will find h applying the algorithm from the proof
- $f = \overbrace{(aa)}^{u}\, a\, \overbrace{(b^3a^3b^3)}^{v} \sim_4 (aa)b\,\overbrace{(b^3a^4b^2)}^{w} = g$ so
- $(aa)a(b^3a^3b^3) \sim_4 (aa)ab(b^3a^4b^2)$ or $(aa)b(b^3a^4b^2) \sim_4 (aa)ba(b^3a^3b^3)$
- $f' = \overbrace{(a^3b^3)}^{u'}\, a\, \overbrace{(a^2b^3)}^{v'} \sim_4 (a^3b^3)b\,\overbrace{(a^4b^2)}^{w'} = g'$

- let $f = a^3b^3a^3b^3$ and $g = a^2b^4a^4b^2$, $f \sim_4 g$
- we will find h applying the algorithm from the proof
- $f = \overbrace{(aa)}^{u}\, a\, \overbrace{(b^3a^3b^3)}^{v} \sim_4 (aa)b\,\overbrace{(b^3a^4b^2)}^{w} = g$ so
- $(aa)a(b^3a^3b^3) \sim_4 (aa)ab(b^3a^4b^2)$ or $(aa)b(b^3a^4b^2) \sim_4 (aa)ba(b^3a^3b^3)$
- $f' = \overbrace{(a^3b^3)}^{u'}\, a\, \overbrace{(a^2b^3)}^{v'} \sim_4 (a^3b^3)b\,\overbrace{(a^4b^2)}^{w'} = g'$
- $(a^3b^3)a(a^2b^3) \sim_4 (a^3b^3)ab(a^4b^2)$ or $(a^3b^3)b(a^4b^2) \sim_4 (a^3b^3)ba(a^2b^3)$

- let $f = a^3b^3a^3b^3$ and $g = a^2b^4a^4b^2$, $f \sim_4 g$
- we will find h applying the algorithm from the proof
- $f = \overbrace{(aa)}^{u}\, a\, \overbrace{(b^3a^3b^3)}^{v} \sim_4 (aa)b\,\overbrace{(b^3a^4b^2)}^{w} = g$ so
- $(aa)a(b^3a^3b^3) \sim_4 (aa)ab(b^3a^4b^2)$ or $(aa)b(b^3a^4b^2) \sim_4 (aa)ba(b^3a^3b^3)$
- $f' = \overbrace{(a^3b^3)}^{u'}\, a\, \overbrace{(a^2b^3)}^{v'} \sim_4 (a^3b^3)b\,\overbrace{(a^4b^2)}^{w'} = g'$
- $(a^3b^3)a(a^2b^3) \sim_4 (a^3b^3)ab(a^4b^2)$ or $(a^3b^3)b(a^4b^2) \sim_4 (a^3b^3)ba(a^2b^3)$

- let $f = a^3b^3a^3b^3$ and $g = a^2b^4a^4b^2$, $f \sim_4 g$
- we will find h applying the algorithm from the proof
- $f = \overbrace{(aa)}^{u}\, a\, \overbrace{(b^3a^3b^3)}^{v} \sim_4 (aa)b\,\overbrace{(b^3a^4b^2)}^{w} = g$ so
- $(aa)a(b^3a^3b^3) \sim_4 (aa)ab(b^3a^4b^2)$ or $(aa)b(b^3a^4b^2) \sim_4 (aa)ba(b^3a^3b^3)$
- $f' = \overbrace{(a^3b^3)}^{u'}\, a\, \overbrace{(a^2b^3)}^{v'} \sim_4 (a^3b^3)b\,\overbrace{(a^4b^2)}^{w'} = g'$
- $(a^3b^3)a(a^2b^3) \sim_4 (a^3b^3)ab(a^4b^2)$ or $(a^3b^3)b(a^4b^2) \sim_4 (a^3b^3)ba(a^2b^3)$
- then $f'' = \overbrace{(a^3b^4a^3)}^{u''}\, a\, \overbrace{(b^2)}^{v''} \sim_4 (a^3b^4a^3)b\,\overbrace{(b^2)}^{w''} = g''$

- let $f = a^3b^3a^3b^3$ and $g = a^2b^4a^4b^2$, $f \sim_4 g$
- we will find h applying the algorithm from the proof
- $f = \overbrace{(aa)}^{u}\,a\,\overbrace{(b^3a^3b^3)}^{v} \sim_4 (aa)b\,\overbrace{(b^3a^4b^2)}^{w} = g$ so
- $(aa)a(b^3a^3b^3) \sim_4 (aa)ab(b^3a^4b^2)$ or $(aa)b(b^3a^4b^2) \sim_4 (aa)ba(b^3a^3b^3)$
- $f' = \overbrace{(a^3b^3)}^{u'}\,a\,\overbrace{(a^2b^3)}^{v'} \sim_4 (a^3b^3)b\,\overbrace{(a^4b^2)}^{w'} = g'$
- $(a^3b^3)a(a^2b^3) \sim_4 (a^3b^3)ab(a^4b^2)$ or $(a^3b^3)b(a^4b^2) \sim_4 (a^3b^3)ba(a^2b^3)$
- then $f'' = \overbrace{(a^3b^4a^3)}^{u''}\,a\,\overbrace{(b^2)}^{v''} \sim_4 (a^3b^4a^3)b\,\overbrace{(b^2)}^{w''} = g''$
- $(a^3b^4a^3)a(b^2) \sim_4 (a^3b^4a^3)ab(b^2)$ or $(a^3b^4a^3)b(b^2) \sim_4 (a^3b^4a^3)ba(b^2)$

- let $f = a^3b^3a^3b^3$ and $g = a^2b^4a^4b^2$, $f \sim_4 g$
- we will find h applying the algorithm from the proof
- $f = \overbrace{(aa)}^{u}\, a\, \overbrace{(b^3a^3b^3)}^{v} \sim_4 (aa)b\, \overbrace{(b^3a^4b^2)}^{w} = g$ so
- $(aa)a(b^3a^3b^3) \sim_4 (aa)ab(b^3a^4b^2)$ or $(aa)b(b^3a^4b^2) \sim_4 (aa)ba(b^3a^3b^3)$
- $f' = \overbrace{(a^3b^3)}^{u'}\, a\, \overbrace{(a^2b^3)}^{v'} \sim_4 (a^3b^3)b\, \overbrace{(a^4b^2)}^{w'} = g'$
- $(a^3b^3)a(a^2b^3) \sim_4 (a^3b^3)ab(a^4b^2)$ or $(a^3b^3)b(a^4b^2) \sim_4 (a^3b^3)ba(a^2b^3)$
- then $f'' = \overbrace{(a^3b^4a^3)}^{u''}\, a\, \overbrace{(b^2)}^{v''} \sim_4 (a^3b^4a^3)b\, \overbrace{(b^2)}^{w''} = g''$
- $(a^3b^4a^3)a(b^2) \sim_4 (a^3b^4a^3)ab(b^2)$ or $(a^3b^4a^3)b(b^2) \sim_4 (a^3b^4a^3)ba(b^2)$

- let $f = a^3b^3a^3b^3$ and $g = a^2b^4a^4b^2$, $f \sim_4 g$
- we will find h applying the algorithm from the proof
- $f = \overbrace{(aa)}^{u}\, a\, \overbrace{(b^3a^3b^3)}^{v} \sim_4 (aa)b\,\overbrace{(b^3a^4b^2)}^{w} = g$ so
- $(aa)a(b^3a^3b^3) \sim_4 (aa)ab(b^3a^4b^2)$ or $(aa)b(b^3a^4b^2) \sim_4 (aa)ba(b^3a^3b^3)$
- $f' = \overbrace{(a^3b^3)}^{u'}\, a\, \overbrace{(a^2b^3)}^{v'} \sim_4 (a^3b^3)b\,\overbrace{(a^4b^2)}^{w'} = g'$
- $(a^3b^3)a(a^2b^3) \sim_4 (a^3b^3)ab(a^4b^2)$ or $(a^3b^3)b(a^4b^2) \sim_4 (a^3b^3)ba(a^2b^3)$
- then $f'' = \overbrace{(a^3b^4a^3)}^{u''}\, a\, \overbrace{(b^2)}^{v''} \sim_4 (a^3b^4a^3)b\,\overbrace{(b^2)}^{w''} = g''$
- $(a^3b^4a^3)a(b^2) \sim_4 (a^3b^4a^3)ab(b^2)$ or $(a^3b^4a^3)b(b^2) \sim_4 (a^3b^4a^3)ba(b^2)$
- $a^3b^4a^4b^2 \sim_4 a^3b^4a^4b^3 = h$

Thank you for your attention!