Lights and Darks of the Star-Free Star

Edward Ochmański & Krystyna Stawikowska

Nicolaus Copernicus University, Toruń, Poland

Introduction:
star may destroy recognizability

In (finitely generated) trace monoids we have:

- ***singletons are in REC***
- ***REC is closed under union***
- ***REC is closed under catenation***
- ***REC is included in RAT***

Corollary: If a **rational** language is **not recognizable**, recognizability was broken by a **star operation**.

History: questions on recognizable star

Rational expressions are often more convenient (e.g. for inductive proofs) than the automata-oriented or algebraic descriptions of recognizable languages

Definition: **Recognizable star is the classical star, working only if its result is recognizable (and undefined otherwise)**

Questions: Does recognizable star produce (jointly with union and catenation) the whole class REC?

History: recognizable stars in free and trace monoids

Questions: Does recognizable star produce (jointly with union and catenation) the whole class REC?

Answer 1 (Kleene): **In finitely generated free monoid YES.**

Any star keeps recognizability!

Answer 2: **In arbitrary trace monoid YES.**

The connected star keeps recognizability and builds the whole REC.

Monoids, languages, free monoids

Monoid $(M,\cdot)$ – a set M with an associative operation (called *product*) and a neutral element ε (called *unit*).

Language in $(M,\cdot)$ – any subset of M.

Atomic languages (atoms) – empty language $\emptyset$
and singletons $\{m\}$, for $m \in M$.

Free Monoid $(A^*,\cdot)$ – set of all finite words (including ε) over an alphabet A, with concatenation.

Operations on languages

Set-theoretical operations:

union $X \cup Y$

intersection $X \cap Y$

difference $X \setminus Y$

complement $X' = M \setminus X$

Algebraic operations:

product: $XY = \{ xy \mid x \in X, y \in Y \}$

power: $X^0 = \{\varepsilon\}$, $X^{n+1} = XX^n$

star: $X^* = \bigcup \{ X^n \mid n=0,1,\ldots \}$

Rational and star-free languages

Rational languages: languages built up from atoms with operations of **union**, **product** and **star**.

Rational expressions: expressions describing such construction.

RAT(*M*): the class of rational subsets of *M*.

Star-free languages: languages built up from atoms with operations of **union**, **product** and **complement**.

Star-free expressions: expressions describing such construction.

SF(*M*): the class of star-free subsets of *M*.

Syntactic monoid, recognizable and aperiodic languages

syntactic congruence $\approx_L \subseteq M \times M$ of a language $L \subseteq M$**:**

$$u \approx_L v \text{ iff } (\forall x,y \in M)\ xuy \in L \Leftrightarrow xvy \in L$$

syntactic monoid of a language $L \subseteq M$**:**

quotient monoid $M_L = M/{\approx_L}$

A language $L \subseteq M$ is:

recognizable (REC) iff its syntactic monoid M_L is finite;

aperiodic (AP) iff its syntactic monoid M_L is finite and.
aperiodic, i.e. $(\exists n)(\forall x \in M_L)\ x^n = x^{n+1}$.

Star-Free = Aperiodic (in A^*)

Theorem (Schützenberger 1965):

In finitely generated free monoids SF(A^*) = AP(A^*)

i.e. *a language is star-free iff it is aperiodic*

Example: The language $(aa)^*$ has syntactic monoid:

	ε	a
ε	ε	a
a	a	ε

It is not aperiodic,
as $a^n \neq a^{n+1}$ for any n.

By Schützenberger Theorem, $(aa)^*$ is not star-free.

Example: $M=\{a,b\}^*$, $L=(ab)^*$

The language $(ab)^*$ is ***star-free***

$$(ab)^* = (b\varnothing' \cup \varnothing' a \cup \varnothing' aa \varnothing' \cup \varnothing' bb \varnothing')'$$

Syntactic monoid of $(ab)^*$

	ε	a	b	ab	ba	0
ε	ε	a	b	ab	ba	0
a	a	0	ab	0	a	0
b	b	ba	0	b	0	0
ab	ab	a	0	ab	0	0
ba	ba	0	b	0	ba	0
0	0	0	0	0	0	0

$a^2 = a^3 = 0$

$b^2 = b^3 = 0$

$(ab)^2 = (ab)^3 = ab$

$(ba)^2 = (ba)^3 = ba$

$\varepsilon^2 = \varepsilon^3 = \varepsilon$

$0^2 = 0^3 = 0$

$n = 2$

Hence $(ab)^*$ is ***aperiodic***

Operation of STAR-FREE STAR

Star-free star operation:

$$L^{\times} = \begin{cases} L^{*} \text{ if } L^{*} \text{ is star-free} \\ \text{undefined otherwise} \end{cases}$$

SFS-expression: **a rational expression, built from atoms with symbols of union, product and star-free star**

SFS-language: **a language built up from atoms with operations of union, product and star-free star**

Question: **SFS = SF ?**

Inclusion $\subseteq$ holds in any monoid, since union, product and star-free star preserve star-freeness.

Inclusion $\supseteq$ is not general; there are monoids in which it does not hold.

Example: $(Z,+)$ – integers with addition

$$\mathrm{SF} = \{ X \subseteq Z \mid X \text{ or } X' \text{ is finite} \}$$

$$\mathrm{SFS} = \{ X \subseteq Z \mid X = Z \text{ or } X \text{ is finite} \}$$

Star-free star in free monoids

Theorem (O/S 2005):

In finitely generated free monoids SFS = SF

Inclusion $\subseteq$ holds in any monoid.

The proof of $\supseteq$ is based on McNaughton/Yamada (1960) construction of regular expressions for automata and uses the Schützenberger Theorem.

Many-sided characterization of Star-Free Trace Languages

$\mathrm{SF}(A^*) = \mathrm{SFS}(A^*)$

Traces and trace languages

Concurrent alphabet – pair (A,I), where A is a finite alphabet, $I \subseteq A \times A$ is a symmetric and irreflexive ***independence*** relation; complement of I, the relation $D = A \times A - I$, is called ***dependency***.

Trace monoid $M = A^*/_I$ – quotient of A^* by the least congruence containing the relation $\{ab \approx ba \mid aIb\}$.

Traces – members of trace monoids.

Trace languages – subset of trace monoids.

Free monoid – trace monoid with $I = \emptyset$.

Lemma on closing product

Flattening of a trace language:

$$\cup T = \{ w \in A^* \mid [w] \in T \}$$

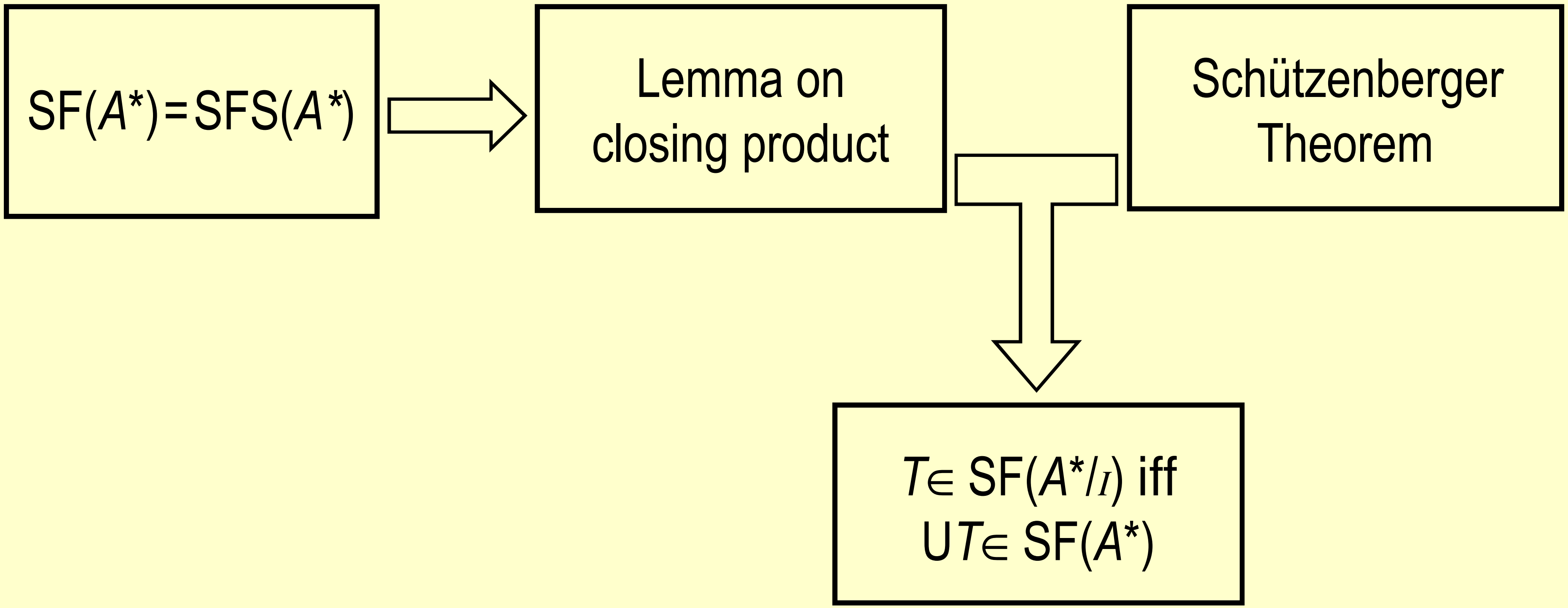
Closure of a word language:

$$\overline{L} = \cup[L] = \{ w \in A^* \mid [w] \in [L] \}$$

L is ***closed*** iff $L = \overline{L}$

Lemat: If $K, L \subseteq A^*$ are closed and star-free, then $\overline{KL}$ is star-free.

Flat characterization of star-free trace languages



Many-sided characterization of Star-Free Trace Languages

$SF(A^*) = SFS(A^*)$ ⟹ $T \in SF(A^*/I)$ iff $\cup T \in SF(A^*)$

Aperiodic characterization of star-free trace languages

Theorem (Guaiana/Restivo/Salemi 1992):

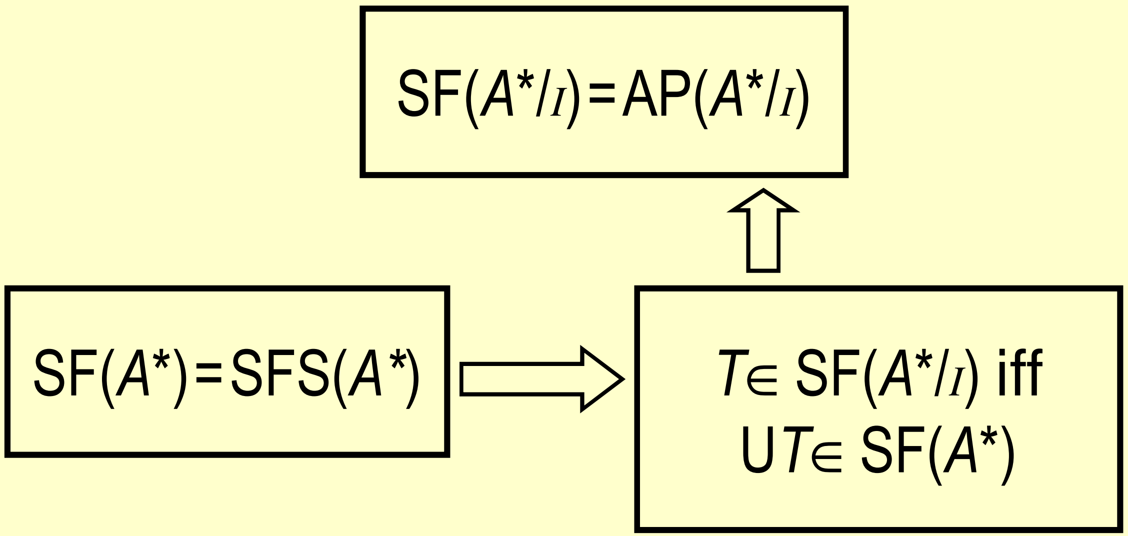
In any trace monoid,

***T* is star-free iff *T* is aperiodic.**

Proof:

$$\begin{array}{ccc} T \in AP(A^*/_I) & \Rightarrow & T \in SF(A^*/_I) \\ \Uparrow & & \Downarrow \\ \cup T \in AP(A^*) & \Leftarrow & \cup T \in SF(A^*) \end{array}$$

Many-sided characterization of Star-Free Trace Languages



Logic for traces (*Thomas 1989*)

(A,I) – concurrent alphabet, $w = a_1 \ldots a_n \in A^*$.

Model for the trace $[w] \in A^*/_I$ is the **trace-graph** $\langle V,E,\lambda \rangle$ where $V=\{x_1 \ldots x_n\}$ – vertices, $E \subseteq V \times V$ – edges and $\lambda : V \to A$ is s.t. $(\forall i)\ \lambda(x_i)=a_i$ and $x_i E x_j$ iff $i<j$ & $a_i D a_j$

First order formula is build up from atomic formulas $x=y$, xEy and $\lambda(x)=a$ with logical connectives and quantifiers.

First order definability

$T(\Psi)=\{\alpha\in A^*/_I \mid \Psi$ is satisfied in $\alpha\}$ – trace language defined by the sentence Ψ.

Trace language T is **first-order definable** iff there is a first-order sentence Ψ s.t. $T=T(\Psi)$.

$FO(A^*/_I)$ – the class of first-order definable languages in the trace monoid $A^*/_I$.

Theorem (McNaughton/Papert 1971):

In finitely generated free monoids $SF(A^*)=FO(A^*)$

Lexicographic representation

$(A,<,I)$ – ordered concurrent alphabet

Lexicographic representative of trace $\alpha \in A^*/I$
– a word $Lex(\alpha) \in A^*$, lexicographically first in α.

$\boldsymbol{Lex(T) = \{ Lex(\alpha) \in A^* \mid \alpha \in T \}}$
– lexicographic representation of the trace language T.

$\boldsymbol{LEX = Lex(A^*/I)}$
– lexicographic representation of the trace monoid.

$LEX = A^* - \bigcup\{A^* b (I_a)^* a A^* \mid aIb \wedge a<b\}$,
where $I_a = \{c \in A \mid aIc\}$, hence ***LEX*** **is star-free**.

Flat and Lex characterizations of first-order trace languages

Theorem (Ebinger/Muscholl 1993):

For any trace language *T*,
the following statements are equivalent:

(1) T is first-order definable in $A^*/_I$

(2) $\cup T$ is first-order definable in A^*

(3) $\mathrm{Lex}(T)$ is first-order definable in A^*

Many-sided characterization of Star-Free Trace Languages

$SF(A^*/I) = AP(A^*/I)$

⇧

$SF(A^*) = SFS(A^*)$ ⇨ $T \in SF(A^*/I)$ iff $UT \in SF(A^*)$

$T \in FO(A^*/I)$ iff
$UT \in FO(A^*)$ iff
$Lex(T) \in FO(A^*)$

Logical characterization of star-free trace languages

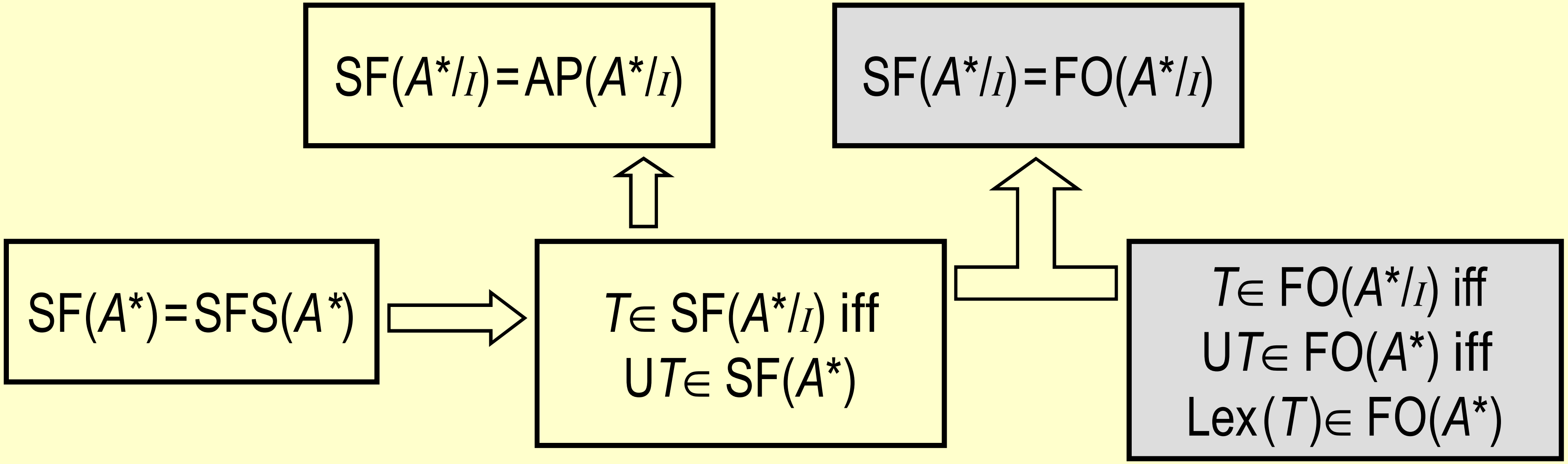
Theorem (Ebinger/Muschol 93, Diekert/Metivier 97):

A trace language is star-free
if and only if
it is first-order definable

Proof. $T \in \mathrm{SF}(A^*/_I)$ iff $\cup T \in \mathrm{SF}(A^*)$ iff

$\cup T \in \mathrm{FO}(A^*)$ iff $T \in \mathrm{FO}(A^*/_I)$

Many-sided characterization of Star-Free Trace Languages



Lexicographic characterization of star-free trace languages

Theorem:

A trace language is star-free in $A^*/_I$

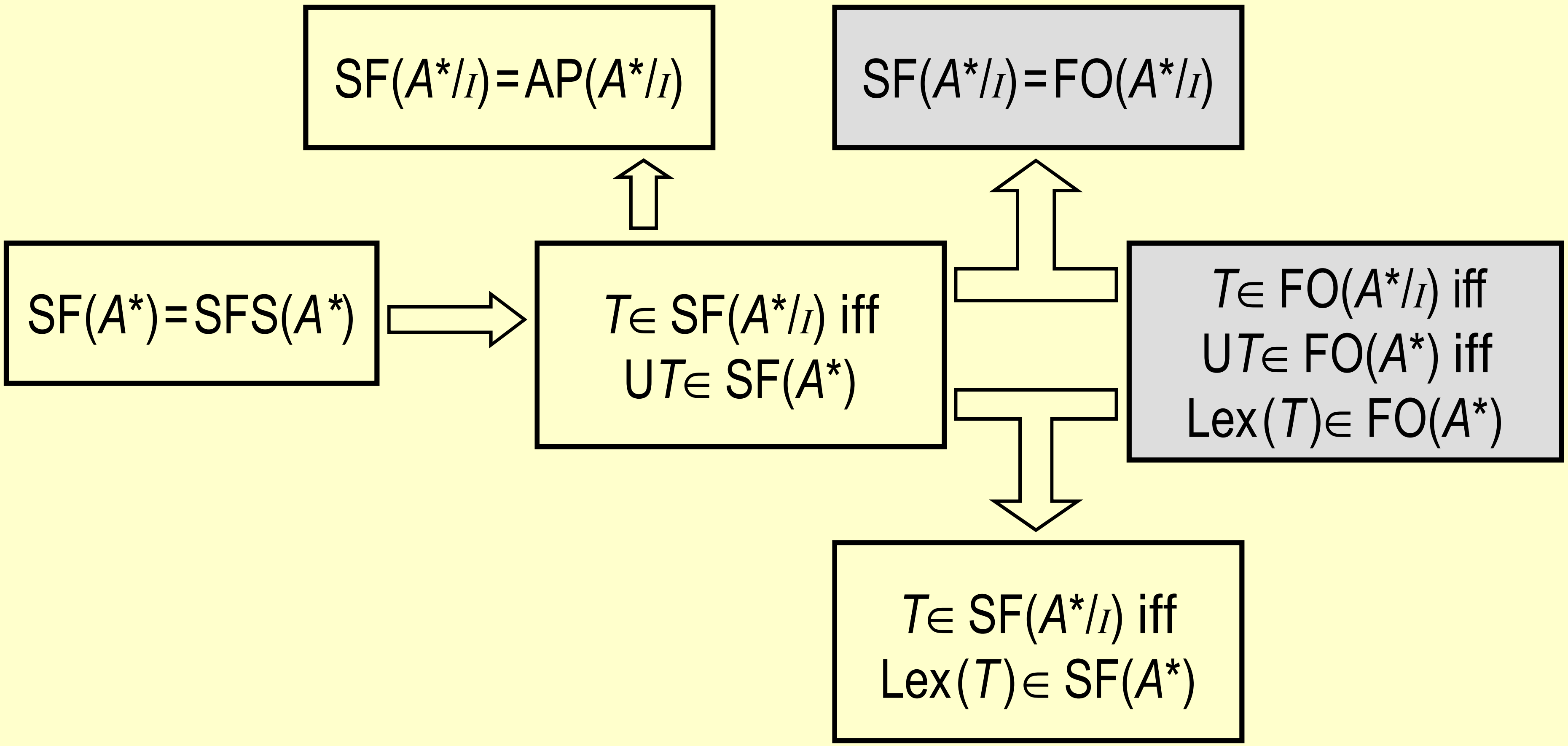
if and only if

its lexicographic representation is star-free in A^*

Proof. $T \in SF(A^*/_I)$ iff $T \in FO(A^*/_I)$ iff

$Lex(T) \in FO(A^*)$ iff $Lex(T) \in SF(A^*)$

Many-sided characterization of Star-Free Trace Languages



Star-free star in trace monoids

Lemma 1: **If $L \in$ SFS(A^*) and $L \subseteq LEX$, then $[L] \in$ SFS($A^*/_I$)**

Proof. Structural induction on SFS-expressions.

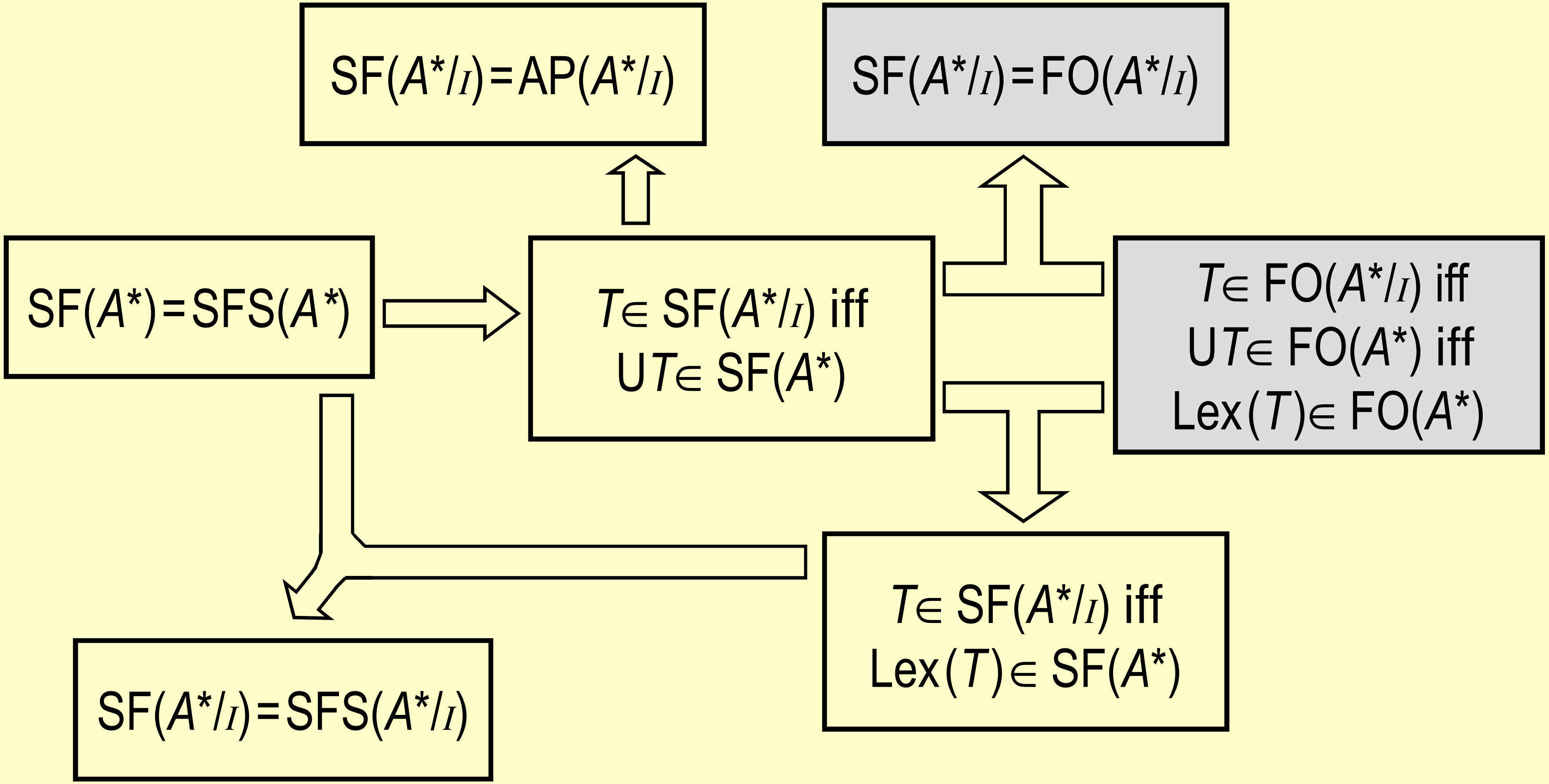
Lemma 2: **If $\mathrm{U}T \in$ SFS(A^*), then $T \in$ SFS($A^*/_I$)**

Theorem: **In trace monoids SFS = SF**

Proof. $T \in \mathrm{SF}(A^*/_I) \Rightarrow \mathrm{U}T \in \mathrm{SF}(A^*) \Rightarrow$

$\mathrm{U}T \in \mathrm{SFS}(A^*) \Rightarrow T \in \mathrm{SFS}(A^*/_I) \Rightarrow T \in \mathrm{SF}(A^*/_I)$

Many-sided characterization of Star-Free Trace Languages



Problems 1-3

Problem 1: **To find syntactically formulated conditions for star-free star in trace monoids.**

Problem 2:

In which monoids SFS=SF? Does it hold in concurrency monoids of Droste?

Decision Problems

Theorem (Muscholl/Petersen 1996):

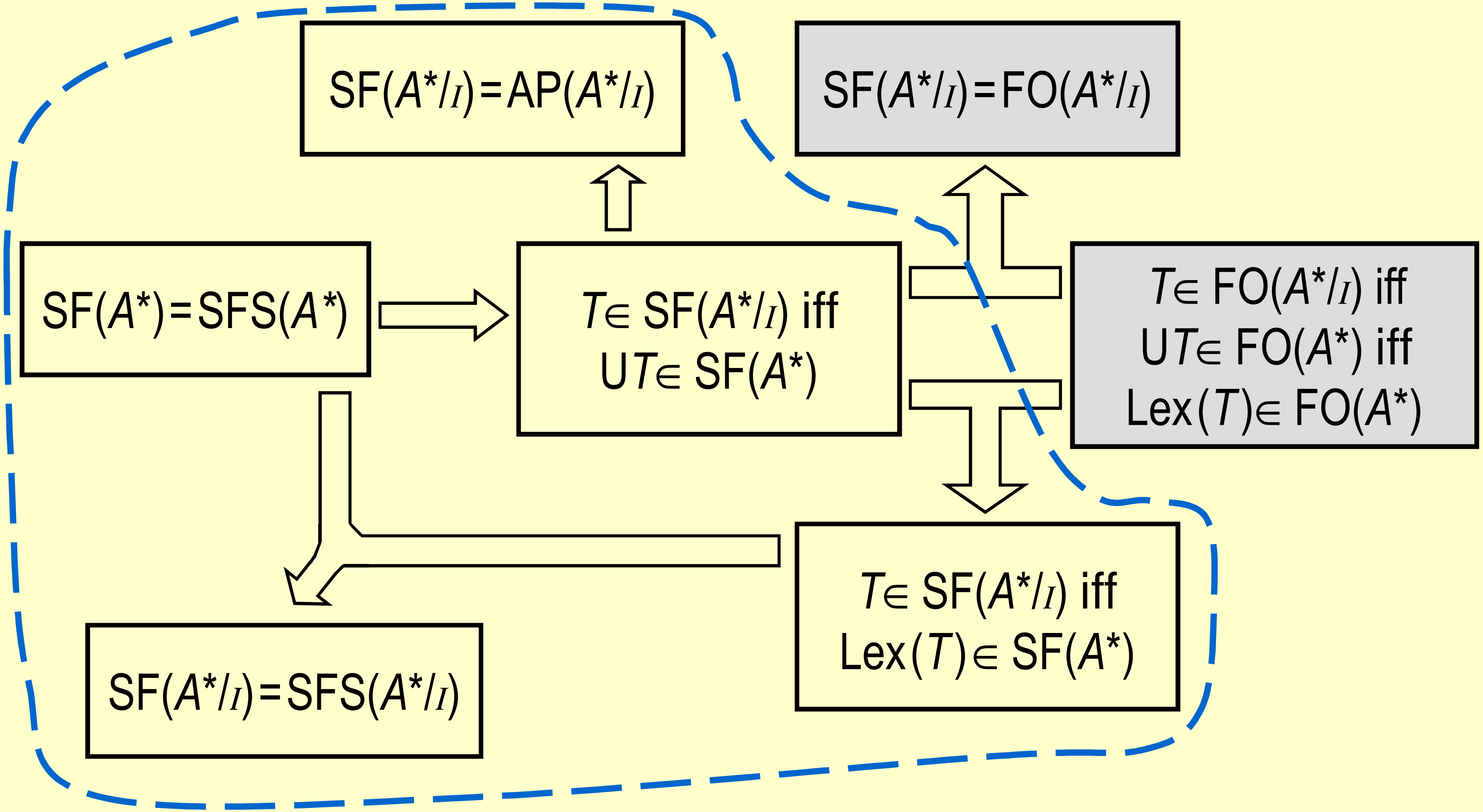
The problem ***"Is T star-free?"*** is decidable for rational trace languages only in trace monoids with transitive independency.

Problem 3 (star problem for s-f trace languages):

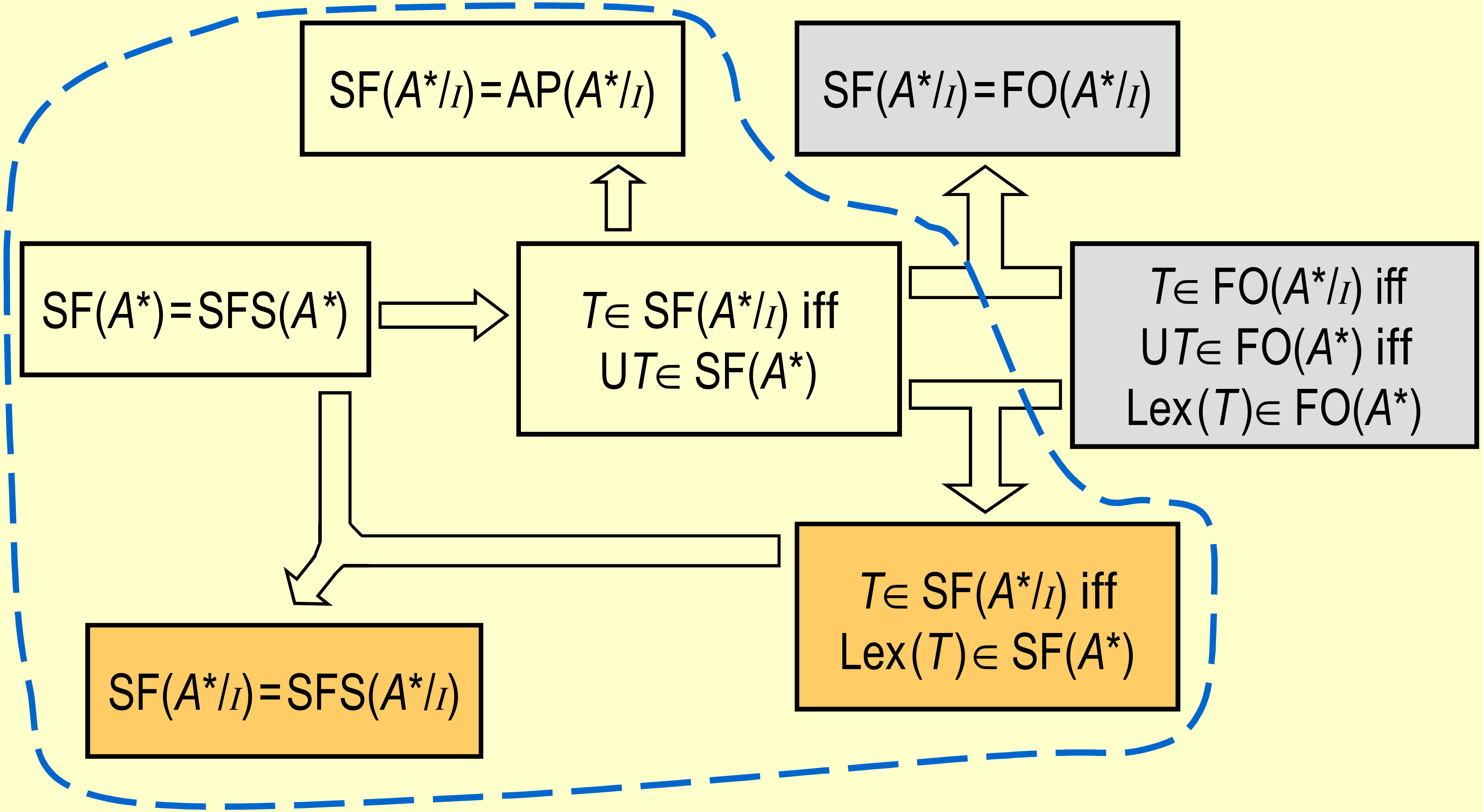
Is the question „Is *T star-free?" decidable for:**

- **star-free trace languages?**
- **finite trace languages?**

A part without logic



A part without logic



Problem 4 – avoiding logic

Problem 4:

To prove the non-logic part of the characterization without logic

LSF languages

lexicographic product:

$$X \circ Y = XY \text{ if } XY \subseteq LEX \text{ else undefined}$$

lexicographic complement:

$$X'' = LEX - X$$

LSF = the class of word languages that are built from atoms with **union**, **lexicographic product** and **lexicographic complement**

Question: *LSF = SF inside LEX ?*

Lemma: **If $L \in LSF$, then $[L] \in SF(A^*/_I)$**

Proof: By lemma on closing product.

By definition: **If $L \in LSF$, then $L \in SF$ and $L \subseteq LEX$**

Question: ***Does the converse hold?***

Transitively oriented alphabets

An ordered alphabet $(A,<,I)$ is **transitively oriented** iff $< \cap I$ is transitive.

Lemma: **If *L*$\in$ *SF and L*$\subseteq$*LEX*, then *L*$\in$ *LSF***

Proposition: ***LSF* = {*L*$\subseteq$*A** | *L*$\in$ *SF and L*$\subseteq$*LEX*}**
if $(A,<,I)$ is transitively oriented

Problem 4a: ***Does the proposition hold for any ordered alphabet?***

Example D: a — c — b

$LEX = (a^*b^*c)^*a^*b^* = a^*b^*(ca^*b^*)^*$

$a^* = LEX - (LEX \cdot c \cdot LEX \cup LEX \cdot b)$

$b^* = LEX - (LEX \cdot c \cdot LEX \cup a \cdot LEX)$

$c^* = LEX - ((LEX - LEX \cdot b) \cdot a \cdot LEX \cup LEX \cdot b \cdot (LEX - a \cdot LEX))$

$(a \cup c)^* = LEX - (LEX \cdot b \cup LEX \cdot bc \cdot LEX)$

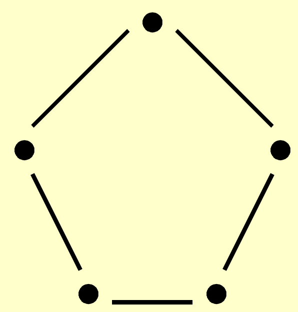
$a^*b^* = LEX - LEX \cdot c \cdot LEX$

$(ac)^* = LEX - (c \cdot LEX \cup (LEX - LEX \cdot b) \cdot a \cup LEX \cdot cc \cdot LEX \cup$
$(LEX - LEX \cdot b) \cdot aa \cdot LEX)$

How is LSF in the Pentagon?

Pentagon cannot be transitively oriented

D:



What about LSF in the pentagon?

Does a* belong to LSF?

Publications

Edward Ochmański, Krystyna Stawikowska: *On Closures of Lexicographic Star-Free Languages*, Proceedings of AFL 2005, pp.227-234, University of Szeged, 2005.

Edward Ochmański, Krystyna Stawikowska: *Star-Free Star and Trace Languages*, Fundamenta Informaticae 72, pp. 323-331, IOS Press 2006.

Krystyna Stawikowska, Edward Ochmański: *On Star-Free Trace Languages and their Lexicographic Representations*, Proceedings of LATA 2007, pp. 541-551, Universitat Rovira i Virgili, Tarragona, Spain, 2007.

Edward Ochmański, Krystyna Stawikowska: *A Star Operation for Star-Free Trace Languages*, Proceedings of DLT 2007, LNCS 4588, pp. 337-345, Springer 2007.

Krystyna Stawikowska: *Word Languages Inducing Recognizable and Star-Free Trace Languages*, PhD Thesis (in Polish), Toruń-Warszawa, 2007.

the end