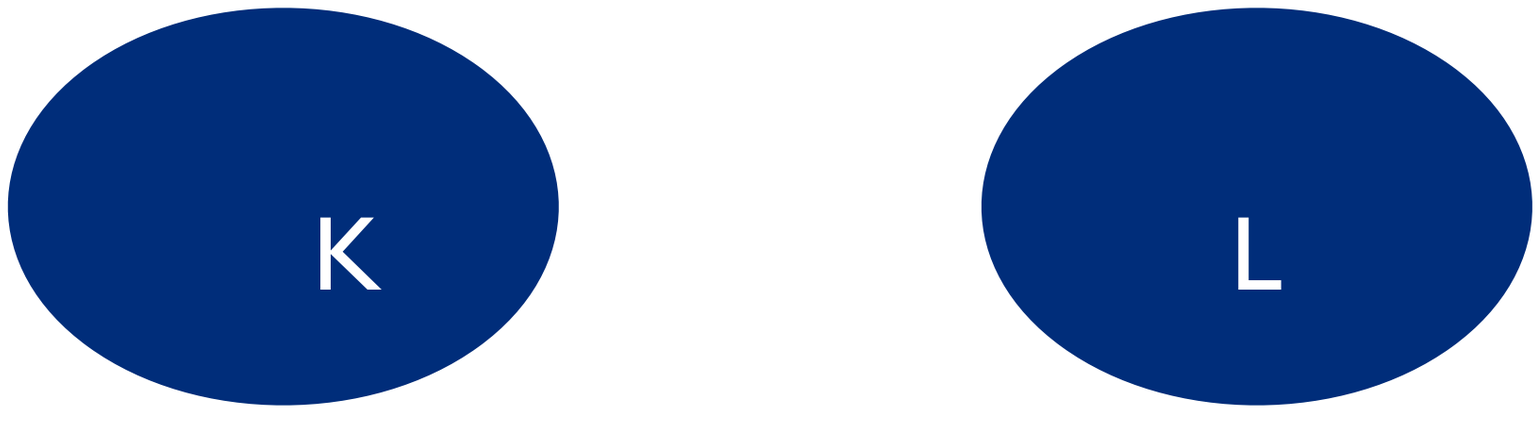
Separability of Context-Free Languages by Piecewise Testable Languages

Wojciech Czerwiński

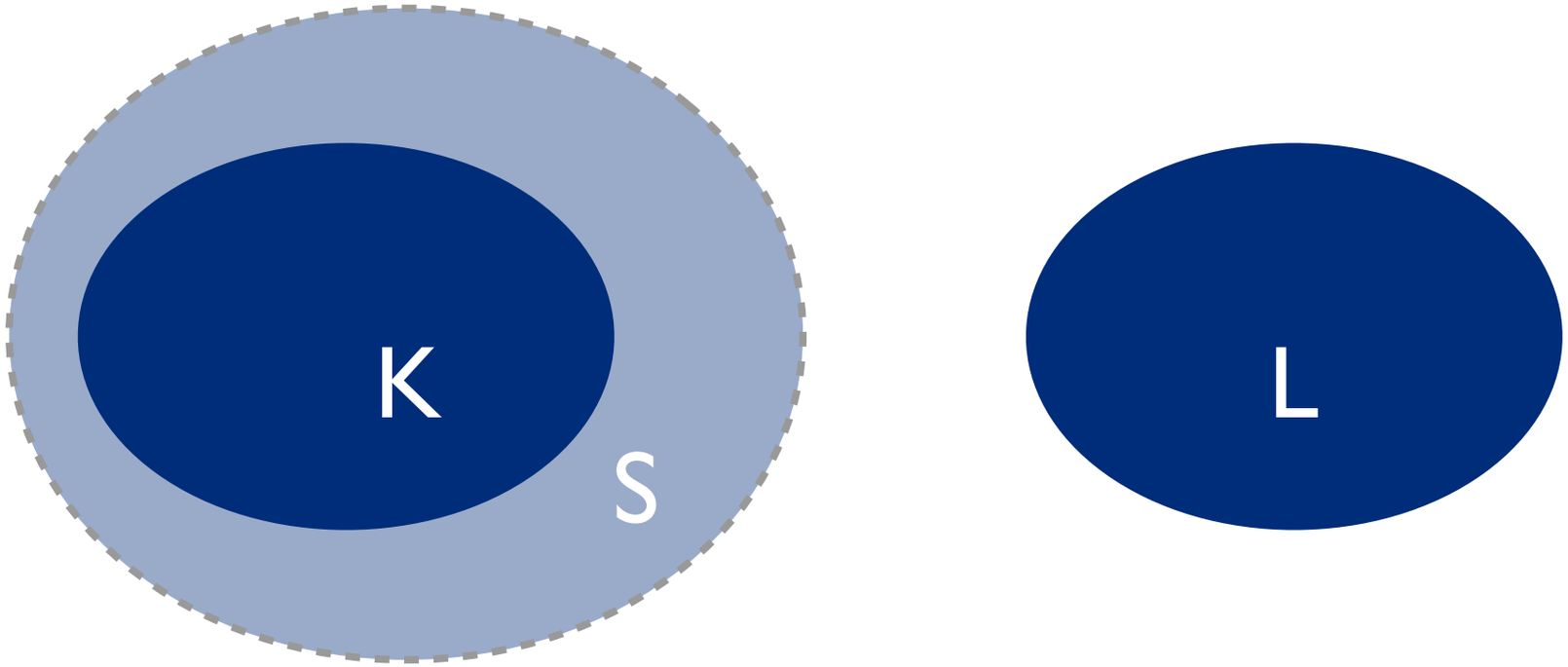
Wim Martens

Separability

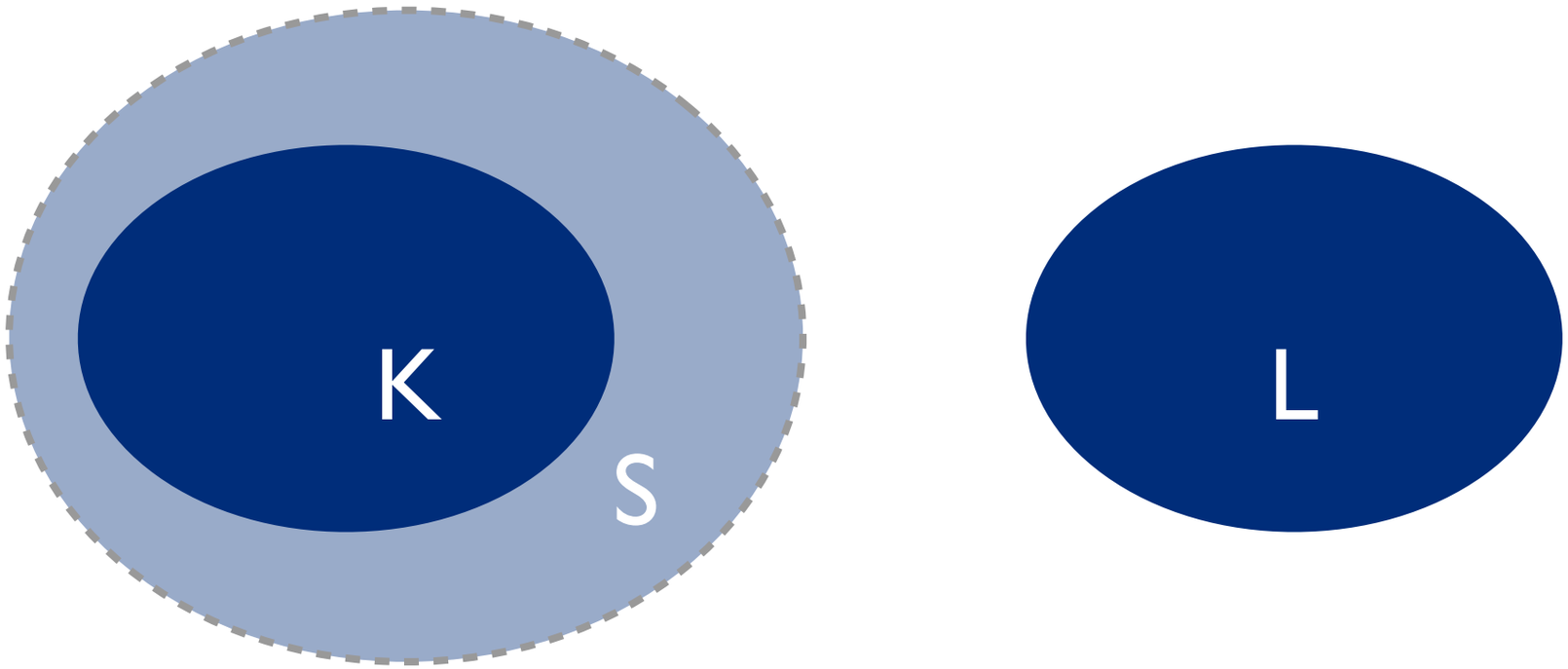
Separability



Separability

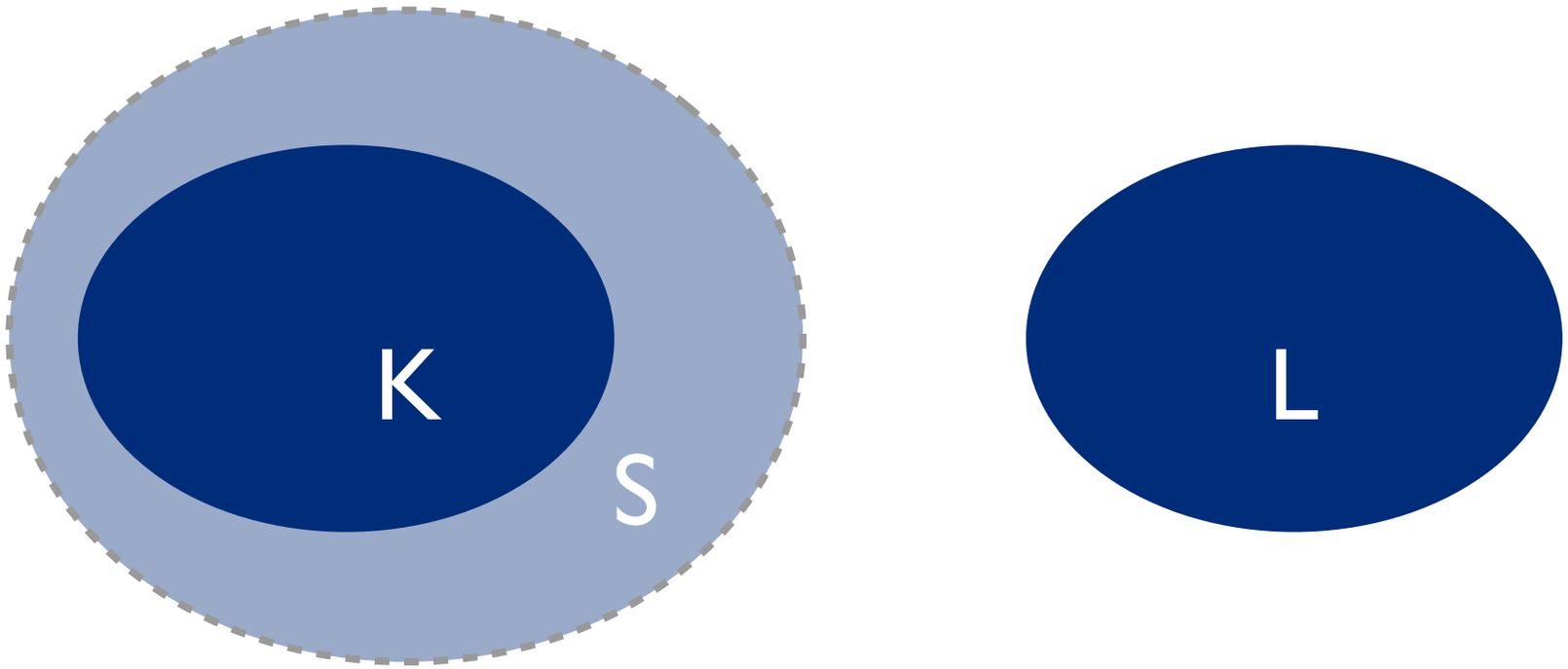


Separability



S *separates* K and L

Separability



S *separates* K and L

K and L are *separable by* family F
if some S from F separates them

Problem

Problem

Given: context-free grammars for languages K and L

Problem

Given: context-free grammars for languages K and L

Question: are K and L separable by piecewise testable languages (PTL)?

Problem

Given: context-free grammars for languages K and L

Question: are K and L separable by piecewise testable languages (PTL)?

piece language

Problem

Given: context-free grammars for languages K and L

Question: are K and L separable by piecewise testable languages (PTL)?

piece language $\Sigma^* a_1 \Sigma^* a_2 \Sigma^* \ldots \Sigma^* a_n \Sigma^*$

Problem

Given: context-free grammars for languages K and L

Question: are K and L separable by piecewise testable languages (PTL)?

piece language $\Sigma^* a_1 \Sigma^* a_2 \Sigma^* \ldots \Sigma^* a_n \Sigma^*$

piecewise testable language

Problem

Given: context-free grammars for languages K and L

Question: are K and L separable by piecewise testable languages (PTL)?

piece language $\Sigma^* a_1 \Sigma^* a_2 \Sigma^* \ldots \Sigma^* a_n \Sigma^*$

piecewise testable language bool. comb. of pieces

What is known?

Separability of CFL by

What is known?

Separability of CFL by

- CFL - undecidable (intersection problem)

What is known?

Separability of CFL by

- CFL - undecidable (intersection problem)
- regular languages - undecidable

What is known?

Separability of CFL by

- CFL - undecidable (intersection problem)
- regular languages - undecidable
- any family containing (reverse)-definite languages - undecidable

Definite languages

Definite languages

reverse definite language = finite union of $w\Sigma^*$

Definite languages

reverse definite language = finite union of $w\Sigma^*$

any logic *L*

Definite languages

reverse definite language = finite union of $w\Sigma^*$

any logic *L*

- able to express n-th letter equals a

Definite languages

reverse definite language = finite union of $w\Sigma^*$

any logic *L*

- able to express n-th letter equals a
- closed under boolean combinations

Definite languages

reverse definite language = finite union of $w\Sigma^*$

any logic *L*

- able to express n-th letter equals a
- closed under boolean combinations

describes all reverse definite languages

Our main result

Our main result

Theorem:

Separability of context free languages
by piecewise testable languages
is decidable

Our main message

Our main message

- something nontrivial possible for separability of CFL

Our main message

- something nontrivial possible for separability of CFL
- no algebra needed

Our main message

- something nontrivial possible for separability of CFL
- no algebra needed
- piecewise testable languages are special

Generalization

Generalization

The same construction works for separating:

Generalization

The same construction works for separating:

- languages of Petri Nets

Generalization

The same construction works for separating:

- languages of Petri Nets
- languages of Lossy Counter Machines (?)

Generalization

The same construction works for separating:

- languages of Petri Nets
- languages of Lossy Counter Machines (?)
- every class of well-behaving languages

Thank you!

Proof (sketch)

Proof (sketch)

Two semi-procedures

Proof (sketch)

Two semi-procedures

One tries to show
separability

Proof (sketch)

Two semi-procedures

One tries to show separability

One tries to show non-separability

Proof (sketch)

Two semi-procedures

One tries to show separability

Enumerates all piecewise testable languages and test them

One tries to show non-separability

Proof (sketch)

Two semi-procedures

One tries to show separability

Enumerates all piecewise testable languages and test them

One tries to show non-separability

Enumerates all patterns and test them

Patterns

Patterns

Pattern p over Σ consists of:

Patterns

Pattern p over Σ consists of:

words $w_0, w_1, ..., w_n$ in Σ^*

Patterns

Pattern p over Σ consists of:

words $w_0, w_1, ..., w_n$ in Σ^*

subalphabets $B_1, ..., B_n$ of Σ

Patterns

Pattern p over Σ consists of:

words $w_0, w_1, ..., w_n$ in Σ^*

subalphabets $B_1, ..., B_n$ of Σ

$B^{\otimes}$ = words from B^* that contain all the letters from B

Patterns

Pattern p over Σ consists of:

words $w_0, w_1, ..., w_n$ in Σ^*

subalphabets $B_1, ..., B_n$ of Σ

$B^{\otimes}$ = words from B^* that contain all the letters from B

Pattern p fits to a language L if
for all $k \geq 0$ intersection of L and
$w_0 (B_1^{\otimes})^k w_1 ... w_{n-1} (B_n^{\otimes})^k w_n$
is nonempty

Patterns and separability

Patterns and separability

Theorem (van Rooijen, Zeitoun `13):
Languages K and L are non-separable by PTL
if and only if
there exists a pattern p,
that fits to both to K and L