Post Correspondence Problem

Simon Schrodi

DHBW Karlsruhe

12.04.2019

Motivation

Given two context-free grammars G_1, G_2, the following decision problems are undecidable:

- $L(G_1) \cap L(G_2) = \emptyset$? (cutting problem)
- $|L(G_1) \cap L(G_2)| = \infty$? (finiteness problem)
- $L(G_1) \subseteq L(G_2)$? (inclusion problem)
- $L(G_1) = L(G_2)$? (equivalence problem)
- Is G_1 ambiguous? (ambiguity problem)
- ...

Repitition: Reduction theorem

Be $A \subseteq \Sigma^*$ and $B \subseteq \Gamma^*$ languages. $A \leq B$ iff there is a total, calculable function $f : \Sigma^* \to \Gamma^*$, so that $\forall x \in \Sigma^*$ the following is valid:

$$x \in A \Leftrightarrow f(x) \in B. \tag{0}$$

If $A \leq B$ and A is undecidable, then B is also undecidable.

Definition: Post Correspondence Problem (PCP)

An instance of PCP consists of a **finite sequence**

$$K = [(x_1, y_1), ..., (x_k, y_k)], \tag{1.1}$$

where $x_i, y_i \neq \epsilon$ is preceded by a finite alphabet Σ. It is to be decided whether there is a **corresponding index sequence**

$$i_1, ..., i_n \in [1, ..., k], n \geq 1, \tag{1.2}$$

also called **solution**, so that the following holds

$$x_{i_1} x_{i_2} ... x_{i_n} = y_{i_1} y_{i_2} ... y_{i_n}. \tag{1.3}$$

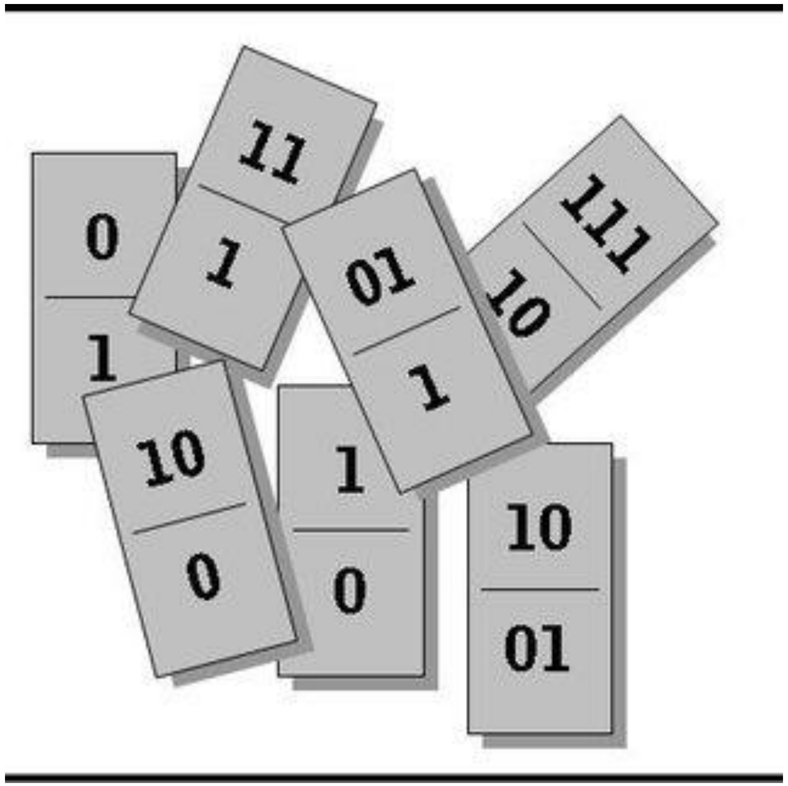


Image source:: https://www.researchgate.net/profile/Klaus_Jantke/publication/297212574/figure/fig2/AS:337034053996545@1457366588223/Abbildung-23-Unloesbare-Problemstellung-Dass-es-fuer-den-Pool-von-Wortpaaren-in-der-Abb_Q320.jpg

Definition: Modified PCP (MPCP)

- Same as PCP with the additional condition $\boldsymbol{i_1 = 1}$

Example PCP 1

Given: $K = [(10111, 10), (1, 111), (10, 0)]$
Searching for: corresponding index sequence

Solution: The index sequence is $(1, 2, 2, 3)$:

$$\underbrace{10111}_{x_1}\underbrace{1}_{x_2}\underbrace{1}_{x_2}\underbrace{10}_{x_3} = 101111110 = \underbrace{10}_{y_1}\underbrace{111}_{y_2}\underbrace{111}_{y_2}\underbrace{0}_{y_3}$$

Example PCP 2

Given: $K = [(10, 101), (011, 11), (101, 011)]$

Solution: There is not matching index sequence (**argument of compulsion to move**)

- Every potential solution must start with $i_1 = 1$
- Whenever the y-sequence has a 1 lead, the only possible continuation of the sequence is:

$$x - sequence: \; ... \; \underbrace{101}_{x_3}$$

$$y - sequence: \; ... \; 1\,\underbrace{011}_{y_3}$$

The y-sequence has always a 1 lead over the x-sequence

Beispiel PKP 3

Given: $K = [(001, 0), (01, 011), (01, 101), (10, 001)]$

Solution: $(2, 4, 3, 4, 4, 2, 1, 2, 4, 3, 4, 3, ...)$ with 66 indices

Theorem: MPCP is semi-decidable

- Combinatorial decision tree
- Depth search or breadth search?

Auxiliary theorem: MPCP is undecidable

Show the following:

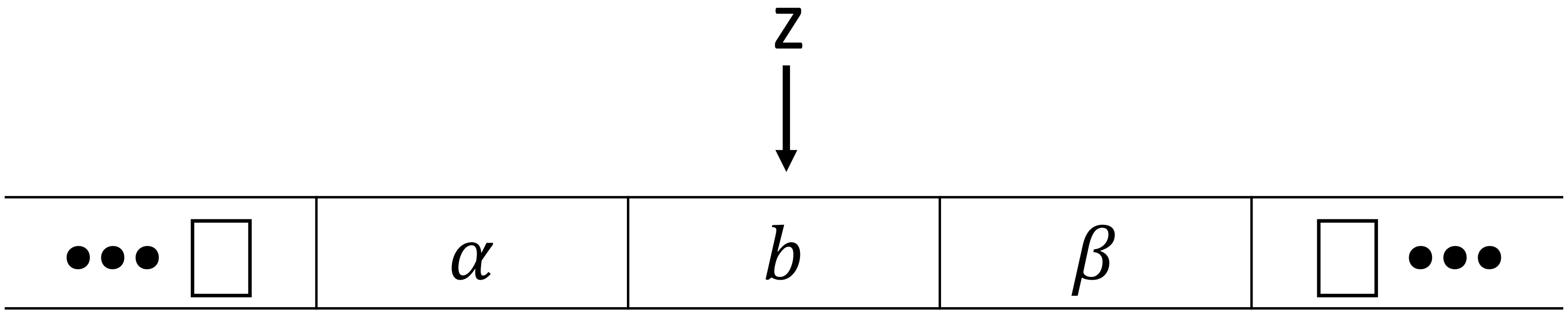
$$H \leq MPKP, \tag{2.1}$$

by giving a reduction function f that maps the input from the general halting problem $H = \{(p, w) | \text{programm p halts when run on input w}\}$ to inputs from MPCP such that:

$$(p, w) \in H(M_p) \Leftrightarrow f(p, w) \in MPKP \tag{2.2}$$

Configuration of a Turing machine M_w

- Configuration of a Turing machine is a word $k = \alpha z b \beta \in \Gamma^* Z \Gamma^*$
- Snapshot of a Turing machine:



Idea of proof

- Present x- and y-sequence as configuration sequences of a TM
- The y-sequence has always a one configuration lead
- The x-sequence catches up after halting of the Turing machine

Exemplary Simulation

Be given a Turing machine M with input word $w = 01$:

$$M = (\underbrace{\{z_1, z_2, z_f\}}_{Z}, \underbrace{\{0, 1\}}_{\Sigma}, \underbrace{\{0, 1, \square\}}_{\Gamma}, \delta, \underbrace{z_1}_{z_1 \in Z}, \square, \underbrace{\{z_f\}}_{E})$$

δ	0	1	$\square$
z_1	$(z_2, 1, R)$	$(z_2, 0, L)$	$(z_2, 1, L)$
z_2	$(z_f, 0, L)$	$(z_1, 0, R)$	$(z_2, 0, R)$
z_f	—	—	—

Exemplary Simulation: Solution 1

- Rule sequence: **S(tarting rule)**

x- und y-sequence as configuration sequence

- x-sequence: #
- y- sequence: $\#\boldsymbol{z_1}\mathbf{01}\#$

y-sequence has always a one configuration lead

- M_w: $\boldsymbol{z_1}\mathbf{01}$

δ	0	1	$\Box$
z_1	$(z_2, 1, R)$	$(z_2, 0, L)$	$(z_2, 1, L)$
z_2	$(z_f, 0, L)$	$(z_1, 0, R)$	$(z_2, 0, R)$
z_f	—	—	—

Exemplary Simulation: Solution 2

- Rule sequence: S,**(R)ight-transfer rule**

- x-sequence: $\#\mathbf{z_1 0}$
- y-sequence: $\#z_1 01\#\mathbf{1 z_2}$

y-sequence has always a one configuration lead

- $M_w: z_1 01 \vdash \mathbf{1 z_2} 1$

δ	0	1	$\square$
z_1	$(z_2, 1, R)$	$(z_2, 0, L)$	$(z_2, 1, L)$
z_2	$(z_f, 0, L)$	$(z_1, 0, R)$	$(z_2, 0, R)$
z_f	—	—	—

Exemplary Simulation: Solution 3

- Rule sequence: S, R, **(C)opy rule**

- x-sequence: $\#z_1 0\mathbf{1}$
- y-sequence: $\#z_1 01\#1z_2\mathbf{1}$

y-sequence has always a one configuration lead

- M_w: $z_1 01 \vdash \mathbf{1z_2 1}$

δ	0	1	$\square$
z_1	$(z_2, 1, R)$	$(z_2, 0, L)$	$(z_2, 1, L)$
z_2	$(z_f, 0, L)$	$(z_1, 0, R)$	$(z_2, 0, R)$
z_f	—	—	—

Exemplary Simulation: Solution 4

- Rule sequence: S, R, C, C

- x-sequence: $\#z_1 01\#$
- y-sequence: $\#z_1 01\#1z_2 1\#$

y-sequence has always a one configuration lead

- $M_w\colon z_1 01 \vdash \mathbf{1z_2 1}$

δ	0	1	$\square$
z_1	$(z_2, 1, R)$	$(z_2, 0, L)$	$(z_2, 1, L)$
z_2	$(z_f, 0, L)$	$(z_1, 0, R)$	$(z_2, 0, R)$
z_f	—	—	—

Exemplary Simulation: Solution 5

- Rule sequence: S, R, C, C, **C**

- x-sequence: $\#z_1 01\#\mathbf{1}$
- y-sequence: $\#z_1 01\#1z_2 1\#\mathbf{1}$

y-sequence has always a one configuration lead

- M_w: $z_1 01 \vdash 1z_2 1 \vdash \mathbf{10}\boldsymbol{z_1}$

δ	0	1	$\square$
z_1	$(z_2, 1, R)$	$(z_2, 0, L)$	$(z_2, 1, L)$
z_2	$(z_f, 0, L)$	$(z_1, 0, R)$	$(z_2, 0, R)$
z_f	—	—	—

Exemplary Simulation: Solution 6

- Rule Sequence: S, R, C, C, C, **R**

- x-Sequenz: $\#z_1 01\#1\boldsymbol{z_2 1}$
- y-Sequenz: $\#z_1 01\#1z_2 1\#1\boldsymbol{0z_1}$

y-sequence has always a one configuration lead

- M_w: $z_1 01 \vdash 1z_2 1 \vdash \mathbf{1}\boldsymbol{0z_1}$

δ	0	1	$\square$
z_1	$(z_2, 1, R)$	$(z_2, 0, L)$	$(z_2, 1, L)$
z_2	$(z_f, 0, L)$	$(z_1, 0, R)$	$(z_2, 0, R)$
z_f	—	—	—

Exemplary Simulation: Solution 7

- Rule sequence: S, R, C, C, C, C, **C**, **C**, **(S1)-special rule**

- x-sequence: $\#z_1 01\#1z_2 1\#\mathbf{10z_1}\#$
- y-sequence: $\#z_1 01\#1z_2 1\#10z_1\#\mathbf{1z_2 01\#}$

y-sequence has always a one configuration lead

- M_w: $z_1 01 \vdash \cdots \vdash 10z_1 \vdash \mathbf{1z_2 01}$

δ	0	1	$\square$
z_1	$(z_2, 1, R)$	$(z_2, 0, L)$	$(z_2, 1, L)$
z_2	$(z_f, 0, L)$	$(z_1, 0, R)$	$(z_2, 0, R)$
z_f	—	—	—

Exemplary Simulation: Solution 8

- Rule sequence: S, R, C, C, C, R, C, C, S1, **(L)eft-transfer rule**, **C**, **C**

- x-sequence: $\#z_1 01\#1z_2 1\#10z_1\#\mathbf{1z_2 0}\mathbf{1}\#$
- y-sequence: $\#z_1 01\#1z_2 1\#10z_1\#1z_2 01\#\boldsymbol{z_f}\mathbf{101}\#$

- M_w: $z_1 01 \ \vdash \cdots \vdash 1z_2 01 \vdash \boldsymbol{z_f}\mathbf{101}$

δ	0	1	$\square$
z_1	$(z_2, 1, R)$	$(z_2, 0, L)$	$(z_2, 1, L)$
z_2	$(z_f, 0, L)$	$(z_1, 0, R)$	$(z_2, 0, R)$
z_f	—	—	—

Exemplary Simulation: Solution 9

- Rule sequence: S, R, C, C, C, R, C, C, S1, L, C, C, **(D)eletion rule**

- x-sequence: $\#z_1 01\#1z_2 1\#10z_1\#1z_2 01\#\boldsymbol{z_f 1}$
- y-sequence: $\#z_1 01\#1z_2 1\#10z_1\#1z_2 01\#z_f 101\#\boldsymbol{z_f}$

x-sequence catches up after halting of Turing machine

Exemplary Simulation: Solution 10

- Rule sequence: S, R, C, C, C, R, C, C, S1, L, C, C, L, **C**, **C**, **C**, **D**, **C**, **C**, **D**, **C**, **(C)losing rule**
- x-sequence: $\#z_1 01\#1z_2 1\#10z_1\#1z_2 01\#z_f 1\mathbf{01}\#\boldsymbol{z_f}\mathbf{01}\#\boldsymbol{z_f}\mathbf{1}\#\boldsymbol{z_f}\#\#$
- y-sequence: $\#z_1 01\#1z_2 1\#10z_1\#1z_2 01\#z_f 101\#z_f\mathbf{01}\#\boldsymbol{z_f}\mathbf{1}\#\boldsymbol{z_f}\#\#$

Mapping rules

(i) **Starting rule:** $(\#, \#z_1 w\#)$

(ii) **Copy rule:** $\forall a \in \Gamma \cup \{\#\} : (a, a)$

(iii) **Transfer rule:** $\forall z \in Z \backslash E; \forall z' \in Z; \forall a, c \in \Gamma \backslash \{\Box\}$:

$$\begin{aligned}
&(za, cz'),\ \ falls\ \delta(z, a) = (z', c, R)\\
&(bza, z'bc),\ \ falls\ \delta(z, a) = (z', c, L),\ \ \forall b \in \Gamma\\
&(z\#, cz'\#),\ \ falls\ \delta(z, \Box) = (z', c, R)\\
&(bz\#, z'bc\#),\ \ falls\ \delta(z, \Box) = (z', c, L),\ \ \forall b \in \Gamma \backslash \{\Box\}
\end{aligned}$$

(iv) **Deletion rule:** $\forall z_f \in E; \forall a \in \Gamma \backslash \{\Box\}$: $(az_f, z_f), (z_f a, z_f)$

(v) **Closing rule:** $\forall z_f \in E : (z_f \#\#, \#)$

Proof of reduction theorem: „=>“

- $(p, w) \in H \Rightarrow f(p, w) \in MPKP$
  - If $(p, w) \in H$, we will eventually receive a solution of the form $(k, k\alpha z_f \beta\#)$ where $z_f \in E, \alpha, \beta \in \Gamma^*$
  - By means of the copy rule and deletion rule the lead $\alpha z_f \beta\#$ can be reduced,
  - until the closing rule is applied, so that $(k' z_f \#\#, k' z_f \#\#)$

Proof of reduction theorem: „<="

- $f(p, w) \in MPKP \Rightarrow (p, w) \in H$
  - It is assumed: $(p, w) \notin H \Rightarrow f(p, w) \notin MPKP$
  - Since no finale state $z_f \in E$ is reached, no deletion rules are applied, and
  - therefore the y-sequence always has a one configuration lead

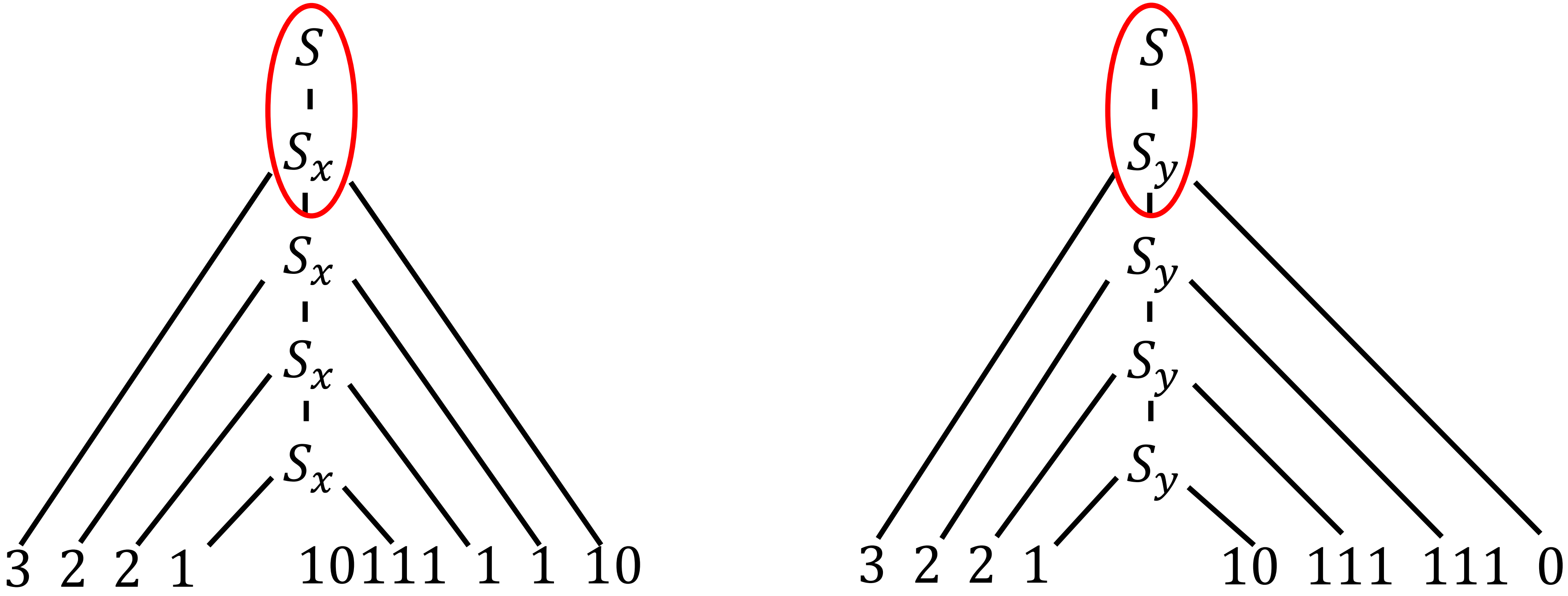
Theorem: Is G ambiguous is undecidable

- Given an instance of the MPCP with K=$[(x_1, y_1), ..., (x_k, y_k)]$ over a finite alphabet Σ and $I = \{i_1, ..., i_k\} \notin \Sigma$
- Construct a context-free grammar$G_x = (V_x, T, P_x, S_x)$ where
  - $T = \Sigma \cup I$
  - $P_x = \{S_x \rightarrow i_1 S_x x_1 | ... | i_k S_x x_k | i_1 x_1\}$

  and a similiar context-free grammar G_y where y_i replace x_i
- Be $L(G_z) = L(G_x) \cup L(G_y)$ where $P_z = \{S \rightarrow S_x | S_y\} \cup P_x \cup P_y$

Theorem: Is G ambiguous is undecidable

- K has a solution $\Rightarrow$ G is ambiguous
  - MPCP of example 1: $K = [(10111, 10), (1, 111), (10, 0)]$ has solution $(1, 2, 2, 3)$

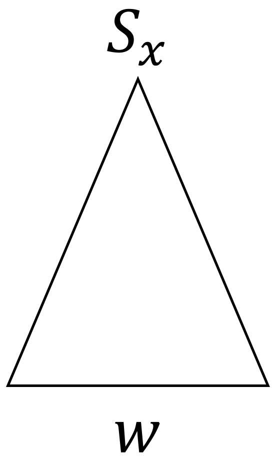


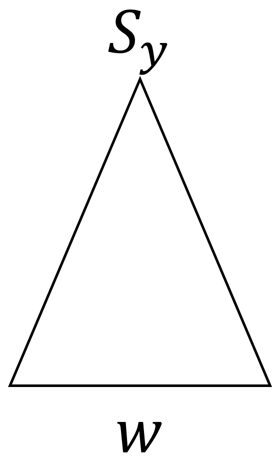
Theorem: Is G ambiguous is undecidable

- G is ambiguous$\Rightarrow$ K has a solution
  - G_z has two **different** syntax trees with the **same** word w
  - Case 1: $S \rightarrow S_x \rightarrow w$ for both syntax trees
    - It follows S_x has to be ambiguous
    - But G_x is $LL(2)$ ↯
  - Case 2: $S \rightarrow S_x \rightarrow w$ respectively $S \rightarrow S_y \rightarrow w$
    - First part of the words are the same (index sequence)
    - Second part of the words are the same $\rightarrow$ Solution for K

Theorem: $L(G_1) \cap L(G_2) \stackrel{?}{=} \emptyset$ is undecidable

- Given an instance of the MPCP with K=$[(x_1, y_1), ..., (x_k, y_k)]$ over a finite alphabet Σ and $I = \{i_1, ..., i_k\} \notin \Sigma$
- Construct context-free grammar $G_x = (V_x, T, P_x, S_x)$ and a similiar contex-free grammar G_y where y_i replace x_i
- $L(G_x) \cap L(G_y) \neq \emptyset \iff \exists w \in \Sigma : w \in L(G_x) \wedge w \in L(G_y)$





Satz: $|L(G_1) \cap L(G_2)| \stackrel{?}{=} \infty$ ist unentscheidbar

- Map two context-free grammars G_x und G_y to MPCP
- Solutions of K correspond to $w \in (L(G_x) \cap L(G_y))$
- If MPCP has at least one solution, then MPCP has infinite many solutio
  by repeating the solution sequence arbitrary times
  - MPCP of example 1: $K = [(10111, 10), (1, 111), (10, 0)]$ hat Lösu
    $(1, 2, 2, 3)$
  - Then (1,2,2,3,1,2,2,3) is also a solution
- Therefore: K has a solution $\iff |L(G_x) \cap L(G_y)| = \infty$

Appendix

Theorem: PCP is undecidable

Show the following:

$$MPKP \leq PKP. \tag{2.0}$$

- Be $K = [(x_1, y_1), ..., (x_k, y_k)]$ an instance of MPCP over finite alphabet Σ
- Be $\#, \$ \notin \Sigma$ two new symbols, so that:

  $$f(K) = [(x'_0, y'_0), (x'_1, y'_1), ..., (x'_k, y'_k), (x'_{k+1}, y'_{k+1})]$$

  where

  - $x'_0 = \#x'_1, x_{k+1} = \$, y'_0 = y'_1, y'_{k+1} = \#\$$
  - $\forall i \in \{1, ..., k\}\ gilt\ x'_i = x_i\#\ bzw.\ y'_i = \#y_i$
- From the reduction function $f(K)$ it is easy to see, that $MPKP \leq PKP$

$MPKP \leq PKP$

Index sequence $(1, 2, 2, 3)$ for MPCP:

$$\underbrace{10111}_{x_1}\underbrace{1}_{x_2}\underbrace{1}_{x_2}\underbrace{10}_{x_3} = 101111110 = \underbrace{10}_{y_1}\underbrace{111}_{y_2}\underbrace{111}_{y_2}\underbrace{0}_{y_3}$$

Index sequence $(0, 2, 2, 3, 4)$ for PCP:

$$\underbrace{\#1\#0\#1\#1\#1\#}_{x'_0}\underbrace{1\#}_{x'_2}\underbrace{1\#}_{x'_2}\underbrace{1\#0\#}_{x'_3}\underbrace{\$}_{x'_4} = \underbrace{\#1\#0}_{y'_0}\underbrace{\#1\#1\#1}_{y'_2}\underbrace{\#1\#1\#1}_{y'_2}\underbrace{\#0}_{y'_3}\underbrace{\#\$}_{y'_4}$$

$MPKP \leq PKP$

Given: $K = [(\underbrace{10111}_{x_1}, \underbrace{10}_{y_1}), (\underbrace{1}_{x_2}, \underbrace{111}_{y_2}), (\underbrace{10}_{x_3}, \underbrace{0}_{y_3})]$.

Therefore

$$f(K) = [(\underbrace{1\#0\#1\#1\#1\#}_{x'_1}\ \underbrace{\#1\#0}_{y'_1}), (\underbrace{1\#}_{x'_2}, \underbrace{\#1\#1\#1}_{y'_2}), \underbrace{\#1\#0}_{y'_2}), (\underbrace{1\#0\#}_{x'_3}, \underbrace{\#0}_{y'_3})]$$

$$\cup[(\underbrace{\#1\#0\#1\#1\#1\#}_{x'_0}, \underbrace{\#1\#0}_{y'_0}), (\underbrace{\#}_{x'_4}, \underbrace{\#\#}_{y'_4})]$$

Exemplary Simulation: Construction 2

Rule tpye	Rule.Index	x-sequence	y-sequence	misc
Starting rule	(i).0	#	$\#z_0 01\#$	
Copy rule	(ii).0	0	0	
	(ii).1	1	1	
	(ii).2	#	#	
Transfer rule	(iii).0	$z_1 0$	$1z_2$	$\delta(z_1, 0) = (z_2, 1, R)$
	(iii).1	$0z_1 1$	$z_2 00$	$\delta(z_1, 1) = (z_2, 0, L)$
	(iii).2	$1z_1 1$	$z_2 10$	$\delta(z_1, 1) = (z_2, 0, L)$
	(iii).3	$0z_1\#$	$z_2 01\#$	$\delta(z_1, \square) = (z_2, 1, L)$
	(iii).4	$1z_1\#$	$z_2 11\#$	$\delta(z_1, \square) = (z_2, 1, L)$
	(iii).5	$0z_2 0$	$z_f 00$	$\delta(z_2, 0) = (z_f, 0, L)$
	(iii).6	$1z_2 0$	$z_f 10$	$\delta(z_2, 0) = (z_f, 0, L)$
	(iii).7	$z_2 1$	$0z_1$	$\delta(z_2, 1) = (z_1, 0, R)$
	(iii).8	$z_2\#$	$0z_2\#$	$\delta(z_2, \square) = (z_2, 0, R)$

Exemplary Simulation: Construction 2

Rule type	Rule.Index	x-sequence	y-sequence	misc
Deletion rule	(iv).0	$0z_f0$	z_f	
	(iv).1	$0z_f1$	z_f	
	(iv).2	$1z_f0$	z_f	
	(iv).3	$1z_f1$	z_f	
	(iv).4	$0z_f$	z_f	
	(iv).5	$1z_f$	z_f	
	(iv).6	z_f0	z_f	
	(iv).7	z_f1	z_f	
Closing rule	(v).0	$z_f\#\#$	$\#$	

Theorem: $L(G_1) \subseteq L(G_2)$ und $L(G_1) = L(G_2)$, are undecideable

- Map two context-free grammars G_x und G_y to MPCP
- L_x and L_y are deterministic context-free
- There exists $\bar{G_x}$ as well as $\bar{L_x}$ and the following holds $L(G_{\bar{x}y}) = L(\bar{G_x}) \cup L(G_y)$
- Assertion: $L(G_x) \cup L(G_y) \stackrel{?}{=} \emptyset \mapsto L(G_{\bar{x}y}) = L(\bar{G_x})$

$$
\begin{aligned}
L(G_x) \cap L(G_y) = \emptyset &\Leftrightarrow L(G_y) \subseteq L(\bar{G_x}) \\
&\Leftrightarrow L(G_y) \cup L(\bar{G_x}) = L(\bar{G_x}) \\
&\Leftrightarrow L(G_{\bar{x}y}) = L_{\bar{G_x}}
\end{aligned}
$$

- Therefore the inclusion problem as well as equivalence problem are undecideable
- But: Equivalence problem is decideable for deterministic context-free grammars!