

Postov korešpondenčný problém

Peter Kostolányi

19. marca 2024

Z minulého semestra je nám známych viacero *nerozhodnuteľných problémov* – predovšetkým treba spomenúť *univerzálny problém*, *diagonálny problém* a *problém zastavenia*. Neexistuje teda žiaden algoritmus – deterministický Turingov stroj zastavujúci na každom vstupe – ktorý by niektorý z týchto problémov riešil. K týmto poznatkom v nasledujúcom pridáme nerozhodnuteľnosť tzv. *Postovho korešpondenčného problému*¹ (PKP).

Všetky nerozhodnuteľné problémy z minulého semestra predpokladajú ako svoj vstup kód nejakého Turingovho stroja; hoci má nerozhodnuteľnosť takýchto problémov množstvo dôležitých implikácií, stále možno – snáď aj oprávnene – namietat, že ide o problémy viac-menej umelé. Postov korešpondenčný problém je naopak formulovaný veľmi jednoducho a čisto kombinatoricky. To umožňuje jeho využitie na dôkaz nerozhodnuteľnosti pomerne širokej škály ďalších problémov.

1 Definícia Postovho korešpondenčného problému

Postov korešpondenčný problém nad abecedou Σ možno na poloformálnej úrovni opísať nasledujúcim spôsobom: na vstupe je daná sada $n \in \mathbb{N} \setminus \{0\}$ druhov „dominových dlaždíc“

$$\begin{array}{|c|} \hline x_1 \\ \hline y_1 \\ \hline \end{array} \begin{array}{|c|} \hline x_2 \\ \hline y_2 \\ \hline \end{array} \cdots \begin{array}{|c|} \hline x_n \\ \hline y_n \\ \hline \end{array};$$

dlaždica k -teho druhu je pritom pre $k = 1, \dots, n$ označená neprázdny slovom $x_k \in \Sigma^+$ na „hornom poschodí“ a neprázdny slovom $y_k \in \Sigma^+$ na „spodnom poschodí“. Z každého druhu je k dispozícii neobmedzene veľa dlaždíc. Úlohou je zistiť, či existuje neprázdna postupnosť prípustných dlaždíc taká, že po ich priložení vedľa seba vznikne po zretážení slov na oboch „poschodiach“ rovnaké výsledné slovo. Dlaždice k sebe možno prikladať iba svojou „dlhšou stranou“ a nemožno ich otáčať. Jednu takúto vstupnú „sadu dlaždíc“ nazývame *prípado* – alebo *inštanciou* – Postovho korešpondenčného problému. *Riešenie* prípadu PKP si možno predstaviť ako postupnosť dlaždíc spĺňajúcu vyššie uvedenú podmienku.

Príklad 1. Uvažujme prípad Postovho korešpondenčného problému nad abecedou $\Sigma = \{a, b\}$ daný nasledujúcou sadou troch dlaždíc:

$$\begin{array}{|c|} \hline aa \\ \hline a \\ \hline \end{array} \begin{array}{|c|} \hline a \\ \hline aa \\ \hline \end{array} \begin{array}{|c|} \hline ba \\ \hline aa \\ \hline \end{array}.$$

Ľahko možno uhádnuť, že tento prípad PKP má riešenie

$$\begin{array}{|c|} \hline aa \\ \hline a \\ \hline \end{array} \begin{array}{|c|} \hline a \\ \hline aa \\ \hline \end{array} :$$

na hornom aj na spodnom „poschodí“ je po zretážení rovnaké slovo aaa . Podobne je riešením aj

$$\begin{array}{|c|} \hline aa \\ \hline a \\ \hline \end{array} \begin{array}{|c|} \hline a \\ \hline aa \\ \hline \end{array} \begin{array}{|c|} \hline aa \\ \hline a \\ \hline \end{array} \begin{array}{|c|} \hline a \\ \hline aa \\ \hline \end{array}$$

alebo

$$\begin{array}{|c|} \hline a \\ \hline aa \\ \hline \end{array} \begin{array}{|c|} \hline a \\ \hline aa \\ \hline \end{array} \begin{array}{|c|} \hline aa \\ \hline a \\ \hline \end{array} \begin{array}{|c|} \hline aa \\ \hline a \\ \hline \end{array}.$$

Ľahko ale tiež napríklad vidieť, že neexistuje žiadne riešenie tohto prípadu PKP obsahujúce tretiu dlaždicu zo vstupu.

¹ Alebo *Postovho problému priradenia*; prívlastok „korešpondenčný“ je odvodený od slova „korešpondencia“ vo význame „zhoda“.

Uvedené definície môžeme sformalizovať tak, že budeme jednotlivé dlaždice považovať za prvky nejakej ďalšej abecedy Δ . Následne môžeme uvažovať dvojicu homomorfizmov $f, g: \Delta^* \rightarrow \Sigma^*$ takých, že pre všetky dlaždice $d \in \Delta$ je $f(d)$ slovo na „hornom poschodí“ dlaždice d a $g(d)$ slovo na „spodnom poschodí“ dlaždice d . Vďaka vlastnosti homomorfizmov je potom pre ľubovoľné slovo $x \in \Delta^+$, reprezentujúce nejakú neprázdnu postupnosť dlaždíc, slovo $f(x)$ rovné zreťazeniu slov na „horných poschodiach“ týchto dlaždíc a slovo $g(x)$ je rovné zreťazeniu slov na ich „spodných poschodiach“. Vo výsledku sa tak iba pýtame, či existuje neprázdne slovo $x \in \Delta^+$ také, že $f(x) = g(x)$.

Postov korešpondenčný problém (PKP) nad abecedou Σ je teda daný nasledovne:

Vstup: Abeceda Δ a nevymazávajúce homomorfizmy $f, g: \Delta^* \rightarrow \Sigma^*$ dané obrazmi jednotlivých písmen.

Výstup: „Áno“ práve vtedy, keď existuje $x \in \Delta^+$ také, že $f(x) = g(x)$.

Bolo by pritom možné obmedziť sa aj na abecedy $\Delta = \{1, \dots, n\}$ pre $n \in \mathbb{N} \setminus \{0\}$; písmená abecedy Δ musia byť v každom prípade podobným spôsobom očíslované pri predávaní inštancie PKP Turingovmu stroju – Turingov stroj totiž vždy pracuje nad pevne danou vstupnou abecedou.

Prípadom alebo *inštanciou* Postovho korešpondenčného problému nad abecedou Σ nazývame trojicu (Δ, f, g) , kde Δ je abeceda a $f, g: \Delta^* \rightarrow \Sigma^*$ sú nevymazávajúce homomorfizmy; ide teda o reprezentáciu jednej sady vstupov. Ľubovoľné $x \in \Delta^+$ také, že $f(x) = g(x)$ potom nazývame *riešením* prípadu (Δ, f, g) . Je ale dôležité nemýliť si existenciu riešenia prípadu PKP s algoritmickou riešiteľnosťou Postovho korešpondenčného problému – algoritmicky riešiť PKP znamená vedieť *pre každý prípad* PKP zistiť, či má riešenie.

Príklad 2. Prípad PKP nad abecedou $\Sigma = \{a, b\}$ z príkladu 1 možno sformalizovať ako trojicu (Δ, f, g) , kde $\Delta = \{1, 2, 3\}$ a $f, g: \Delta^* \rightarrow \Sigma^*$ sú homomorfizmy dané ako

$$\begin{aligned} f(1) &= aa, & f(2) &= a, & f(3) &= ba, \\ g(1) &= a, & g(2) &= aa, & g(3) &= aa. \end{aligned}$$

Jeho riešeniami sú tak napríklad slová $x_1 = 12$, $x_2 = 1212$, alebo $x_3 = 2211$.

Postov korešpondenčný problém *je* zjavne *rekurzívne vyčísliteľný* pre všetky abecedy Σ – stačí postupne skúšať všetky postupnosti dlaždíc zo vstupu a akceptovať, ak niektorá z týchto postupností zodpovedá riešeniu. Dokážeme, že Postov korešpondenčný problém *nie je rozhodnuteľný* pre žiadnu aspoň dvojprvkovú abecedu Σ – neexistuje teda žiaden algoritmus, ktorý by pre daný prípad PKP vedel rozhodnúť, či má riešenie.

2 Modifikovaný Postov korešpondenčný problém

Než ale dokážeme nerozhodnuteľnosť Postovho korešpondenčného problému, dokážeme obdobný výsledok pre jeho variant, tzv. *modifikovaný Postov korešpondenčný problém* (MPKP). Ten potom využijeme na dôkaz nerozhodnuteľnosti samotného PKP.

Modifikovaný Postov korešpondenčný problém (MPKP) nad abecedou Σ je daný nasledovne:

Vstup: Abeceda $\Delta = \{1, \dots, n\}$ pre $n \in \mathbb{N} \setminus \{0\}$ a nevymazávajúce homomorfizmy $f, g: \Delta^* \rightarrow \Sigma^*$ dané obrazmi jednotlivých písmen.

Výstup: „Áno“ práve vtedy, keď existuje $x \in 1\Delta^*$ také, že $f(x) = g(x)$.

Prípad MPKP a jeho riešenie definujeme obdobne ako pre PKP. Pri interpretácii pomocou „dominových dlaždíc“ možno riešenia prípadu MPKP chápať ako riešenia príslušného prípadu PKP, ktoré navyše začínajú prvou dlaždicou. Každá sada dlaždíc určuje prípad PKP aj prípad MPKP – od riešenia prípadu MPKP ale vyžadujeme o niečo viac, než od riešenia prípadu PKP.

3 Nerozhodnuteľnosť modifikovaného PKP

Dokážeme najprv, že modifikovaný PKP je nerozhodnuteľný nad jednou konkrétnou päťprvkovou abecedou. Priamym dôsledkom tohto tvrdenia je – ako ukážeme neskôr – nerozhodnuteľnosť MPKP nad ľubovoľnou aspoň dvojprvkovou abecedou.

Veta 1. *Modifikovaný Postov korešpondenčný problém nad $\Sigma = \{0, 1, \#, \$, Q\}$ je nerozhodnuteľný.*

Dôkaz. Označme ako L_{MPKP} jazyk reprezentujúci modifikovaný Postov korešpondenčný problém nad abecedou Σ . Redukciou problému zastavenia na MPKP nad Σ dokážeme, že MPKP nad Σ nie je rozhodnuteľný – a teda $L_{\text{MPKP}} \notin \mathcal{L}_{\text{rec}}$.

Vstupmi problému zastavenia sú kód $\langle A \rangle$ deterministického stroja A nad binárnou vstupnou abecedou $\{0, 1\}$ a slovo $w \in \{0, 1\}^*$. Budeme pritom predpokladať, že slovo w je neprázdne. Problém očividne zostane nerozhodnuteľný aj po takejto úprave – na dôkaz stačí ku každému Turingovmu stroju A skonštruovať Turingov stroj A' , ktorý „odignoruje“ prvé písmeno vstupu a so zvyškom vstupu pracuje rovnako ako stroj A . Výstup problému zastavenia na vstupoch $\langle A \rangle$ a w je potom rovnaký ako na vstupoch $\langle A' \rangle$ a $0w$, pričom slovo $0w$ je neprázdne.

Za účelom jednoduchšieho ošetrovania okrajových prípadov tiež budeme uvažovať *normálny tvar* Turingových strojov, v ktorom sa čítacia hlava nikdy nehýbe doľava z najľavejšieho zapísaného políčka pásky. Ak teda hlava číta „blank“, vždy ide o „blank“ napravo od zapísanej časti pásky (za predpokladu neprázdnoty vstupu je na páske vždy aspoň jedno zapísané políčko). Dôkaz, že skutočne ide o normálny tvar, prenechávame čitateľovi ako jednoduché cvičenie. Z neho by mala vyplývať aj skutočnosť, že transformáciu do opísaného normálneho tvaru možno realizovať algoritmicke. Preto možno bez ujmy na všeobecnosti predpokladať, že všetky Turingove stroje na vstupe problému zastavenia sú v tomto tvare – problém zostane aj naďalej nerozhodnuteľný.

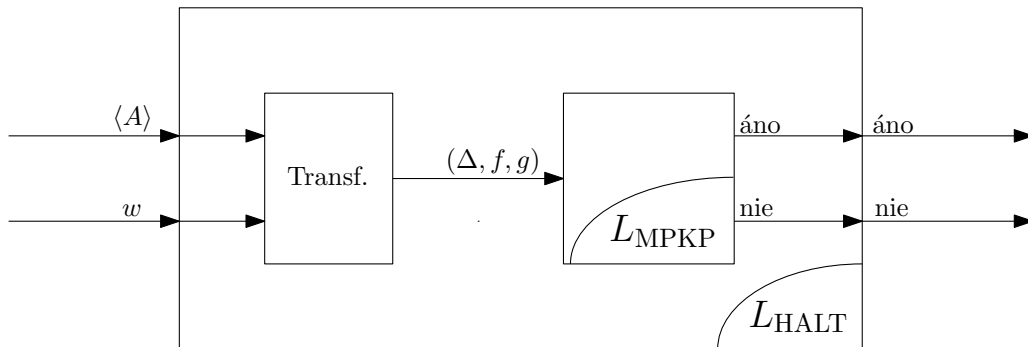
Vo zvyšku dôkazu sa budeme držať týchto predpokladov bez toho, aby sme na to zakaždým upozorňovali.

Predpokladajme teda za účelom sporu, že modifikovaný Postov korešpondenčný problém nad abecedou $\Sigma = \{0, 1, \#, \$, Q\}$ je rozhodnuteľný – čiže $L_{\text{MPKP}} \in \mathcal{L}_{\text{rec}}$. Ukážeme, že v takom prípade musí byť rozhodnuteľný aj problém zastavenia (spor).

Skonštruujeme deterministický Turingov stroj, ktorý sa na každom vstupe zastaví a ktorý akceptuje jazyk L_{HALT} (zodpovedajúci nášmu variantu problému zastavenia). Vstupom takého stroja je kód $\langle A \rangle$ nejakého deterministického Turingovho stroja A nad vstupnou abecedou $\{0, 1\}$ a slovo $w \in \{0, 1\}^+$. Tieto vstupy treba akceptovať práve vtedy, keď sa výpočet stroja A na slove w zastaví.

Stroj rozhodujúci problém zastavenia môže pri rozhodovaní použiť stroj pre MPKP. Prerobíme teda vstupy problému zastavenia na vhodný prípad MPKP tak, aby stroj pre problém zastavenia mohol nejakým spôsobom využiť výstup stroja pre MPKP.

Za týmto účelom zjavne postačí algoritmicke prerobiť vstupy $\langle A \rangle$ a w problému zastavenia na prípad (Δ, f, g) modifikovaného Postovho korešpondenčného problému nad Σ taký, že (Δ, f, g) má riešenie práve vtedy, keď sa výpočet stroja A na w zastaví. Schéma takejto redukcie je znázornená na obrázku 1.



Obr. 1: Základná schéma redukcie problému zastavenia na MPKP nad abecedou Σ .

Zostáva teda nájsť spôsob, ako pre dané $\langle A \rangle$ a w zostrojiť zodpovedajúci prípad (Δ, f, g) . Idea konštrukcie bude spočívať v simulácii výpočtu stroja $A = (K, \Sigma, \Gamma, \delta, q_0, F)$ na slove w pomocou postupne prikladaných dlaždíc tak, že na oboch „poschodiach“ sa budú vytvárať platné postupnosti jeho konfigurácií. Počas simulácie výpočtu pritom bude horné „poschodie“ vždy o jednu konfiguráciu pred spodným „poschodím“. Táto situácia je schematicky znázornená na obrázku 2.

#	C_0	#	C_1	#	C_2	#	$\dots$	#	C_i	#	C_{i+1}	#
#	C_0	#	C_1	#	C_2	#	$\dots$	#	C_i	#		

Obr. 2: Simulácia výpočtu stroja A na jednotlivých „poschodiach“ vytváraných postupne prikladanými dlaždiciami. Konfigurácia C_0 je počiatočná konfigurácia stroja A na slove w a pre $j = 0, \dots, i$ je $C_j \vdash_A C_{j+1}$. „Prečnievajúcu“ konfiguráciu C_{i+1} možno využiť na zabezpečenie nadväznosti jednotlivých konfigurácií, ako uvidíme neskôr pri samotnej konštrukcii dlaždíc.

Špeciálny tvar jednotlivých dlaždíc bude zabezpečovať, aby jednotlivé po sebe idúce konfigurácie boli navzájom v relácii $\vdash_A$. V prípade, že sa na hornom „poschodí“ vyskytne konfigurácia, z ktorej nemožno vo výpočte pokračovať, bude mať spodné „poschodie“ možnosť „dorovnať horné poschodie“, z čoho vyplýva, že daný prípad MPKP bude mať riešenie. To ale bude možné skutočne *iba* v prípade ukončenia výpočtu stroja A , čo znamená, že ak sa stroj A na slove w nezastaví, zodpovedajúci prípad MPKP riešenie mať nebude.

Budeme pracovať s definíciou konfigurácie ako slova, v ktorom stav určuje pozíciu hlavy. Teda napríklad $\mathbf{B}q_0w\mathbf{B}$ je počiatočná konfigurácia stroja A na slove w . „Pravý“ symbol $\mathbf{B}$ je v tejto definícii konfigurácie nadbytočný a píše sa iba z estetických dôvodov. Normálny tvar z úvodu tohto dôkazu však umožňuje vypustiť aj „ľavý“ symbol $\mathbf{B}$. Preto budeme pracovať s konfiguráciami bez ohraničujúcich symbolov $\mathbf{B}$. Počiatočná konfigurácia stroja A na slove w teda bude q_0w .

Treba upozorniť na jeden dôležitý detail: konštruovať postupnosti takto definovaných konfigurácií stroja A je zjavne nemožné, keďže máme k dispozícii iba kód stroja $\langle A \rangle$, a teda iba binárne kódy jednotlivých stavov. Ak by sme v definícii konfigurácie nahradili symbol q priamo jeho binárnym kódom $\langle q \rangle$, došlo by k „pomiešaniu“ symbolov a stavov. Preto budeme v konfiguráciách namiesto symbolu pre stav q uvažovať reťazec $\$ \langle q \rangle \$$, kde $\langle q \rangle$ je binárny kód stavu q . Podobná situácia nastane aj so symbolmi pracovnej abecedy, ktoré ale môžeme pri vhodnom kódovaní nahradiť priamo ich kódom. Namiesto počiatočnej konfigurácie stroja A na slove w teda budeme pracovať s reťazcom $\$ \langle q_0 \rangle \$ \langle w \rangle$. Pri takejto reprezentácii už zjavne nemôže dôjsť k nejednoznačnostiam.² Na oddeľovanie jednotlivých konfigurácií bude slúžiť ďalší špeciálny symbol $\#$.

Môžeme teraz pristúpiť ku konštrukcii jednotlivých dlaždíc prípadu MPKP zodpovedajúceho Turingovmu stroju s kódom $\langle A \rangle$ a slovu w :

Prvá dlaždica: Riešenie prípadu MPKP sa musí začínať prvou dlaždicou. To využijeme na to, aby sme zabezpečili, že sa postupnosť konfigurácií vytváraná postupným prikladaním dlaždíc bude začínať počiatočnou konfiguráciou: ako prvú dlaždicu vezmeme

$$\frac{\# \$ \langle q_0 \rangle \$ \langle w \rangle \#}{\#}.$$

Počiatočná konfigurácia stroja A je pre pevne dané slovo w iba jedna, preto možno túto dlaždicu na základe kódu $\langle A \rangle$ stroja A a slova w bez problémov zostrojiť. *Všetkých možných* konfigurácií stroja A je však nekonečne veľa, čo znamená, že pre ďalšie konfigurácie už nebudeme môcť mať samostatné dlaždice, ale budeme musieť zabezpečiť ich vytváranie po častiach.

²Alternatívne by bolo možné pracovať s kódovaním Turingových strojov, pri ktorom nemôžu mať stavy a pracovné symboly rovnaké kódy. V takom prípade sa dá zaoberať aj bez symbolov $\$$.

Skupina 2: Každá ďalšia konfigurácia vznikne z predchádzajúcej konfigurácie iba lokálnou zmenou poblíž výskytu čítacej hlavy, pričom zvyšok konfigurácie sa iba skopíruje. Druhá skupina dlaždíc bude obsahovať práve takéto „kopírovacie dlaždice“:

$$\begin{array}{|c|} \hline \langle c_1 \rangle \\ \hline \langle c_1 \rangle \\ \hline \end{array} \cdots \begin{array}{|c|} \hline \langle c_m \rangle \\ \hline \langle c_m \rangle \\ \hline \end{array} \begin{array}{|c|} \hline \# \\ \hline \# \\ \hline \end{array},$$

kde $\Gamma = \{c_1, \dots, c_m\}$. „Prečnievajúca konfigurácia na hornom poschodí“ pritom bude zabezpečovať, že skutočne pôjde o kopírovanie predchádzajúcej konfigurácie a nie o pridávanie ľubovoľných symbolov.

Skupina 3: Dlaždice z tejto skupiny budú simulovať vlastný krok výpočtu – čiže lokálnu zmenu v konfigurácii, ktorá sa udeje v okolí políčka čítaného čítacou hlavou. Pre všetky $p, q \in K$ a všetky $a, b \in \Gamma$ také, že $\delta(p, a) = (q, b, 0)$ bude táto skupina obsahovať dlaždicu

$$\begin{array}{|c|} \hline \$\langle q \rangle \$\langle b \rangle \\ \hline \$\langle p \rangle \$\langle a \rangle \\ \hline \end{array}.$$

Podobne pre všetky $p, q \in K$ a všetky $a, b \in \Gamma$ také, že $\delta(p, a) = (q, b, 1)$ bude obsahovať dlaždicu

$$\begin{array}{|c|} \hline \langle b \rangle \$\langle q \rangle \$ \\ \hline \$\langle p \rangle \$\langle a \rangle \\ \hline \end{array},$$

a pre všetky $p, q \in K$ a všetky $a, b \in \Gamma$ také, že $\delta(p, a) = (q, b, -1)$ bude pre všetky $c \in \Gamma$ obsahovať „dlaždicu“

$$\begin{array}{|c|} \hline \$\langle q \rangle \$\langle cb \rangle \\ \hline \langle c \rangle \$\langle p \rangle \$\langle a \rangle \\ \hline \end{array}.$$

V poslednom prípade si vystačíme s dlaždicami tohto typu vďaka dohovoru o normálnom tvare z úvodu dôkazu. Uvedené dlaždice tak riešia všetky situácie, keď hlava číta pracovný symbol. Zostáva doriešiť prípady, keď hlava číta „blank“ – ten musí byť vďaka spomínanému normálnemu tvaru napravo od zapísanej časti pásky.

Pre všetky $p, q \in K$ a $b \in \Gamma$ také, že $\delta(p, \mathbf{B}) = (q, b, 0)$ teda pridáme dlaždicu

$$\begin{array}{|c|} \hline \$\langle q \rangle \$\langle b \rangle \# \\ \hline \$\langle p \rangle \$\# \\ \hline \end{array}.$$

Podobne pre všetky $p, q \in K$ a $b \in \Gamma$ také, že $\delta(p, \mathbf{B}) = (q, b, 1)$ pridáme dlaždicu

$$\begin{array}{|c|} \hline \langle b \rangle \$\langle q \rangle \$\# \\ \hline \$\langle p \rangle \$\# \\ \hline \end{array}$$

a pre všetky $p, q \in K$ a $b \in \Gamma$ také, že $\delta(p, \mathbf{B}) = (q, b, -1)$ pridáme pre všetky $c \in \Gamma$ dlaždicu

$$\begin{array}{|c|} \hline \$\langle q \rangle \$\langle cb \rangle \# \\ \hline \langle c \rangle \$\langle p \rangle \$\# \\ \hline \end{array}.$$

Skupina 4: V prípade, že na hornom „poschodí“ vznikne konfigurácia, z ktorej vo výpočte nemožno pokračovať ďalej, bude mať spodné „poschodie“ možnosť „dorovnať sa“. To však očividne nie je možné s použitím jedinej dlaždice, keďže posledná konfigurácia môže byť ľubovoľne dlhá. Preto je nutné v takejto situácii započat samostatnú fázu prikladania dlaždíc, ktorá sa bude vyznačovať tým, že kód stavu v konfigurácii bude nahradený špeciálnym symbolom Q . Ďalej sa už teda nebudú vytvárať konfigurácie v pravom slova zmysle, ale iba akési „kvázi konfigurácie“. Dlaždice štvrtej skupiny zabezpečujú nahradenie kódu stavu symbolom Q v prípade, že v simulácii výpočtu nemožno ďalej pokračovať. Pre všetky $q \in K$ a $c \in \Gamma$ také, že $\delta(q, c) = \emptyset$ tak bude táto skupina obsahovať „dlaždicu“

$$\begin{array}{|c|} \hline Qc \\ \hline \$\langle q \rangle \$c \\ \hline \end{array}$$

a pre všetky $q \in K$ také, že $\delta(q, \mathbf{B}) = \emptyset$ bude obsahovať dlaždicu

$$\begin{array}{|c|} \hline Q\# \\ \hline \$\langle q \rangle \$\# \\ \hline \end{array}.$$

Skupina 5: Dlaždice tejto skupiny budú umožňovať samotné „dorovnávanie“ oboch postupností konfigurácií tým, že symbol Q bude mať možnosť „mazať“ symboly vo svojom okolí. To znamená, že po konfigurácii, v ktorej sa stroj A zastaví, bude na oboch „poschodiach“ nasledovať postupnosť „kvázi konfigurácií“, kde každá vznikne zmazaním jedného pracovného symbolu z predchádzajúcej. Na konci tohto procesu vznikne „kvázi konfigurácia“ obsahujúca iba Q .

Pre každý pracovný symbol $c \in \Gamma$ teda bude piata skupina obsahovať dlaždice

$$\begin{array}{|c|} \hline Q \\ \hline Q\langle c \rangle \\ \hline \end{array} \quad \begin{array}{|c|} \hline Q \\ \hline \langle c \rangle Q \\ \hline \end{array}.$$

Posledná dlaždica: Dlaždica

$$\begin{array}{|c|} \hline \# \\ \hline Q\#\# \\ \hline \end{array}$$

umožní „dorovnanie oboch poschodí“ v prípade, že na hornom „poschodí“ vznikla „kvázi konfigurácia“ obsahujúca iba symbol Q .

Z uvedených zdôvodnení by malo byť zrejmé, že takto skonštruovaný prípad MPKP má riešenie práve vtedy, keď sa výpočet stroja A na vstupe w zastaví. Navyše možno ľahko nahliadnuť, že transformáciu kódu $\langle A \rangle$ a slova w na opísanú sadu dlaždíc možno realizovať algoritmicky. Tvrdenie je teda dokázané. $\square$

Dôsledok 1. *Nech Σ je aspoň dvojprvková abeceda. Potom je modifikovaný Postov korešpondenčný problém nad abecedou Σ nerozhodnuteľný.*

Dôkaz. Symboly $0, 1, \#, \$, Q$ možno zakódovať do ľubovoľnej abecedy Σ obsahujúcej aspoň dva symboly a, b napríklad ako $0 \mapsto aaa$, $1 \mapsto aab$, $\# \mapsto aba$, $\$ \mapsto abb$ a $Q \mapsto baa$. Je zrejmé, že prípad MPKP nad pôvodnou päťprvkovou abecedou má riešenie práve vtedy, keď má riešenie prípad MPKP nad abecedou Σ , ktorý vznikne zakódovaním oboch „poschodí“ všetkých jeho dlaždíc. Keby teda bol MPKP nad nejakou aspoň dvojprvkovou abecedou Σ rozhodnuteľný, bol by rozhodnuteľný aj MPKP nad abecedou $\{0, 1, \#, \$, Q\}$ – čo by bolo v spore s predchádzajúcou vetou. $\square$

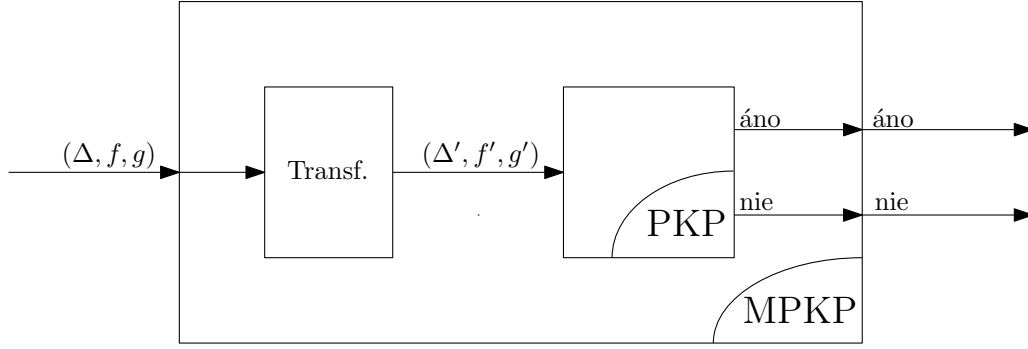
4 Nerozhodnuteľnosť PKP (štandardná redukcia MPKP na PKP)

Vyššie sme dokázali nerozhodnuteľnosť modifikovaného Postovho korešpondenčného problému nad ľubovoľnou aspoň dvojprvkovou abecedou. Teraz využijeme túto skutočnosť na dôkaz nerozhodnuteľnosti „obyčajného“ Postovho korešpondenčného problému.

Veta 2. *Postov korešpondenčný problém nad abecedou $\Sigma = \{0, 1, \#\}$ je nerozhodnuteľný.*

Dôkaz. Redukciou MPKP nad binárnou abecedou na PKP nad Σ . Za účelom sporu predpokladajme, že je PKP nad abecedou Σ rozhodnuteľný. Ukážeme, že v takom prípade by bol rozhodnuteľný aj MPKP nad binárnou abecedou (spor).

Schéma takejto redukcie je znázornená na obrázku 3. Turingov stroj rozhodujúci MPKP nad binárnou abecedou dostane na vstupe prípad MPKP (Δ, f, g) . Ten algoritmicky transformuje na vhodný prípad (Δ', f', g') PKP nad abecedou Σ taký, že (Δ', f', g') má riešenie ako prípad PKP práve vtedy, keď (Δ, f, g) má riešenie ako prípad MPKP. Prípad PKP (Δ', f', g') následne rozhodne pomocou stroja pre PKP, ktorého existenciu predpokladáme.



Obr. 3: Schéma redukcie MPKP nad binárnou abecedou na PKP nad abecedou Σ .

Zostáva opísať konštrukciu prípadu (Δ', f', g') . Nech $w \in \{0, 1\}^+$ je slovo také, že $w = a_1 a_2 \dots a_m$ pre nejaké $m \in \mathbb{N} \setminus \{0\}$ a $a_1, \dots, a_m \in \{0, 1\}$. Ako $\bar{w}$ potom označíme slovo $\bar{w} = a_1 \# a_2 \# \dots \# a_m$.

Pri konštrukcii prípadu (Δ', f', g') využijeme skutočnosť, že ak sa má niektorá dlaždica vyskytovať v riešení ako prvá, musia sa obidve jej „poschodia“ začínať rovnakým písmenom. Dlaždice z pôvodnej sady pre MPKP preto upravíme vhodne pridanými symbolmi $\#$ tak, že túto vlastnosť bude spĺňať iba modifikácia prvej dlaždice z pôvodnej sady, pričom ostatné dlaždice sa budú môcť vyskytovať v riešení až za ňou. Rovnakú vlastnosť síce bude mať aj novopridaná „ukončovacia dlaždica“, ale tá sa na začiatku riešenia evidentne nebude môcť vyskytovať.

Ak je teda prvou dlaždicou pôvodného prípadu MPKP

$$\begin{array}{|c|} \hline x_1 \\ \hline y_1 \\ \hline \end{array},$$

bude zodpovedajúci prípad PKP obsahovať dve dlaždice prislúchajúce k tejto pôvodnej dlaždici – a to

$$\begin{array}{|c|} \hline \# \bar{x}_1 \# \\ \hline \# \bar{y}_1 \\ \hline \end{array}; \quad \begin{array}{|c|} \hline \bar{x}_1 \# \\ \hline \# \bar{y}_1 \\ \hline \end{array};$$

prvou z nich sa bude musieť začínať každé prípadné riešenie konštruovaného prípadu PKP. Pre všetky ostatné dlaždice

$$\begin{array}{|c|} \hline x \\ \hline y \\ \hline \end{array}$$

pôvodného prípadu MPKP bude nová sada pre PKP obsahovať jednu k nej prislúchajúcu dlaždicu

$$\begin{array}{|c|} \hline \bar{x} \# \\ \hline \# \bar{y} \\ \hline \end{array},$$

ktorá sa na začiatku riešenia vyskytovať nemôže. Nakoniec ešte pridáme „ukončovaciu dlaždicu“

$$\begin{array}{|c|} \hline \# \\ \hline \# \# \\ \hline \end{array},$$

ktorou sa každé riešenie konštruovaného prípadu PKP bude musieť končiť.

Čitateľ iste ľahko dokáže, že prípad PKP (Δ', f', g') má riešenie práve vtedy, keď má riešenie prípad MPKP (Δ, f, g) . □

Dôsledok 2. *Nech Σ je aspoň dvojprvková abeceda. Potom je Postov korešpondenčný problém nad abecedou Σ nerozhodnuteľný.*

Dôkaz. Rovnako ako pri dôsledku 1. □

Rovnakú redukciu ako pri dôkaze vety 2 možno aplikovať aj na redukciu MPKP nad ľubovoľnou abecedou Σ na PKP nad abecedou $\Sigma \cup \{\#\}$, kde $\#$ je nový symbol. Túto redukciu budeme nazývať *standardnou redukciou* MPKP na PKP.

5 Riešená úloha

Úloha 1. Uvažujme prípad MPKP pozostávajúci z dlaždíc

aab	ab	bba	ab	a
aa	bab	bb	a	aab

Zapíšte tento prípad formálne a *štandardnou konštrukciou* zostrojte prípad PKP, ktorý má riešenie práve vtedy, keď ho má uvedený prípad MPKP.

Riešenie. Daný prípad MPKP nad abecedou $\Sigma = \{a, b\}$ možno formálne zapísať ako (Δ, f, g) , kde $\Delta = \{1, 2, 3, 4, 5\}$ a homomorfizmy $f, g: \Delta^* \rightarrow \Sigma^*$ sú dané ako

$$\begin{aligned} f(1) &= aab, & f(2) &= ab, & f(3) &= bba, & f(4) &= ab, & f(5) &= a, \\ g(1) &= aa, & g(2) &= bab, & g(3) &= bb, & g(4) &= a, & g(5) &= aab. \end{aligned}$$

Zodpovedajúcim prípadom PKP bude (Δ, f, g) , kde $\Delta = \{1, 2, 3, 4, 5, 6, 7\}$ a homomorfizmy $f, g: \Delta^* \rightarrow (\Sigma \cup \{\#\})^*$ sú dané ako

$$\begin{aligned} f(1) &= \#a\#a\#b\#, & f(2) &= a\#a\#b\#, & f(3) &= a\#b\#, & f(4) &= b\#b\#a\#, \\ f(5) &= a\#b\#, & f(6) &= a\#, & f(7) &= \#, \\ g(1) &= \#a\#a, & g(2) &= \#a\#a, & g(3) &= \#b\#a\#b, & g(4) &= \#b\#b, \\ g(5) &= \#a, & g(6) &= \#a\#a\#b, & g(7) &= \#\#. \end{aligned}$$

Tento prípad PKP teda pozostáva z dlaždíc

$\#a\#a\#b\#$	$a\#a\#b\#$	$a\#b\#$	$b\#b\#a\#$	$a\#b\#$	$a\#$	$\#$
$\#a\#a$	$\#a\#a$	$\#b\#a\#b$	$\#b\#b$	$\#a$	$\#a\#a\#b$	$\#\#$

□