Undecidability of the Post Correspondence Problem

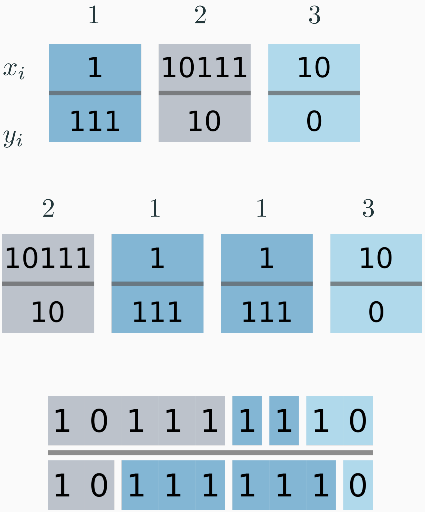
Initial Bachelor Seminar Talk

Edith Heiter

Advisors: Prof. Dr. Gert Smolka, Yannick Forster

March 10, 2017

Post Correspondence Problem



PCP instance

- finite alphabet Σ
- finite set of ordered pairs $\left\{ \left[\frac{x_1}{y_1} \right], \left[\frac{x_2}{y_2} \right], \ldots, \left[\frac{x_k}{y_k} \right] \right\}$
- nonempty strings $x_i, y_i \in \Sigma^+$

Solution of a PCP instance with k cards

- Sequence of indices $i_1, i_2, \ldots, i_m \in \{1, 2, \ldots, k\}$
- $x_{i_1} x_{i_2}, \ldots, x_{i_m} = y_{i_1} y_{i_2}, \ldots, y_{i_m}$

Proof of Undecidability

Theorem

It is undecidable to determine whether a PCP instance has a match.

$PCP = \{\langle P \rangle \mid \text{P is a PCP instance with a match}\}$
$A_{TM} = \{\langle M, w \rangle \mid \text{M is a TM and M accepts w }\}$

$$\mathbf{A_{TM} \leq_m PCP}$$

Many-one reduction

A_{TM} is many-one reducible to PCP, if there exists a computable function f where for every TM M and input w,

$$\langle M, w \rangle \in A_{TM} \Leftrightarrow f(\langle M, w \rangle) \in PCP$$

Reduction Idea $A_{TM} \leq PCP$

TM $M = (Q, \Gamma, \delta, q_0, F)$

$$\frac{\#C_0\#C_1\#\ldots\#C_{n-1}\ldots}{\#C_0\#C_1\#\ldots\#C_{n-1}\#C_n}$$

- a PCP match describes an accepting configuration sequence of M on input w
- define start, transition, and copy-dominos
- transition $\delta(q_0, w_0) = (q_1, a, R)$ corresponds to $\left[\frac{\mathbf{q_0}w_0}{a\mathbf{q_1}}\right]$

$$\left[\frac{\#}{\#q_0w_0w_1\ldots w_n\#}\right] \left[\frac{q_0w_0}{aq_1}\right] \left[\frac{w_1}{w_1}\right] \ldots \left[\frac{w_n}{w_n}\right] \left[\frac{\#}{\#}\right]$$

$$\frac{\#q_0w_0w_1\ldots w_n\#}{\#q_0w_0w_1\ldots w_n\#aq_1w_1\ldots w_n\#}$$

- lower row is one step ahead until a final state is reached
  $\rightarrow$ need to fix the first card of the match

Proof of Undecidability

$A_{TM} = \{\langle M, w\rangle \mid \text{M is a TM and M accepts w }\}$
$PCP = \{\langle P\rangle \mid \text{P is a PCP instance with a match}\}$
$MPCP = \{\langle P\rangle \mid \text{P is a PCP instance with a match that starts with the first domino}\}$

$$A_{TM} \leq_m PCP$$

$$A_{TM} \leq_m MPCP \qquad MPCP \leq_m PCP$$

Reduction MPCP $\leq_m$ PCP

1	2	3
1	10111	10
111	10	0

For string $u = u_1 u_2 \dots u_n \in \Sigma^+$ define

$$\star u = \#u_1\#u_2\#\dots\#u_n$$

$$u\star = u_1\#u_2\#\dots\#u_n\#$$

$$\star u\star = \#u_1\#u_2\#\dots\#u_n\#$$

#1	#1#0#1#1#1	#1#0	#1	#$
1#1#1#	1#0#	0#	#1#1#1#	$

$$P = \left\{ \begin{bmatrix} x_1 \\ \hline y_1 \end{bmatrix}, \begin{bmatrix} x_2 \\ \hline y_2 \end{bmatrix}, \dots, \begin{bmatrix} x_k \\ \hline y_k \end{bmatrix} \right\}$$

$$f(P) = \left\{ \begin{bmatrix} \star x_1 \\ \hline y_1\star \end{bmatrix}, \begin{bmatrix} \star x_2 \\ \hline y_2\star \end{bmatrix}, \dots, \begin{bmatrix} \star x_k \\ \hline y_k\star \end{bmatrix} \right\} \cup \left\{ \begin{bmatrix} \star x_1 \\ \hline \star y_1\star \end{bmatrix} \right\} \cup \left\{ \begin{bmatrix} \#\$ \\ \hline \$ \end{bmatrix} \right\}$$

$$P \in MPCP \Leftrightarrow f(P) \in PCP$$

Reduction MPCP $\leq_m$ PCP

$P \in MPCP \Rightarrow f(P) \in PCP$

$P = \left\{ \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}_1, \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}_2, \ldots, \begin{bmatrix} x_k \\ y_k \end{bmatrix}_k \right\}$ $f(P) = \left\{ \begin{bmatrix} \star x_1 \\ y_1 \star \end{bmatrix}_1, \begin{bmatrix} \star x_2 \\ y_2 \star \end{bmatrix}_2, \ldots, \begin{bmatrix} \star x_k \\ y_k \star \end{bmatrix}_k, \begin{bmatrix} \star x_1 \\ \star y_1 \star \end{bmatrix}_{k+1}, \begin{bmatrix} \#\$ \\ \$ \end{bmatrix}_{k+2} \right\}$

MPCP instance P with solution sequence $(i_1, i_2, \ldots, i_m)$ and $i_1 = 1$

$x_1 x_{i_2} x_{i_3}, \ldots, x_{i_m} =$

$y_1\, y_{i_2}\, y_{i_3}, \ldots, y_{i_m}$

$\Rightarrow f(P)$ has solution $(k+1, i_2, i_3, \ldots, i_m, k+2)$

interleaves all symbols and does not affect the match:

$\star x_1 \star x_{i_2} \star x_{i_3} \ldots \star x_{i_m} \#\$ =$

$\star y_1 \star y_{i_2} \star y_{i_3} \star \ldots\, y_{i_m} \star \$.$

Reduction MPCP $\leq_m$ PCP

$f(P) \in PCP \Rightarrow P \in MPCP$

$P = \left\{ \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}_1, \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}_2, \ldots, \begin{bmatrix} x_k \\ y_k \end{bmatrix}_k \right\} \quad f(P) = \left\{ \begin{bmatrix} \star x_1 \\ y_1 \star \end{bmatrix}_1, \begin{bmatrix} \star x_2 \\ y_2 \star \end{bmatrix}_2, \ldots, \begin{bmatrix} \star x_k \\ y_k \star \end{bmatrix}_k, \begin{bmatrix} \star x_1 \\ \star y_1 \star \end{bmatrix}_{k+1}, \begin{bmatrix} \#\$ \\ \$ \end{bmatrix}_{k+2} \right\}$

PCP instance $f(P)$ with solution $(i_1, i_2, \ldots, i_m) \in \{1, \ldots, k+2\}$
assume $i_2, i_3, \ldots, i_{m-1} \in \{1, \ldots, k\}$

$\begin{bmatrix} \star x_1 \\ \star y_1 \star \end{bmatrix} \begin{bmatrix} \star x_{i_2} \\ y_{i_2} \star \end{bmatrix} \ldots \begin{bmatrix} \star x_{i_{m-1}} \\ y_{i_{m-1}} \star \end{bmatrix} \begin{bmatrix} \#\$ \\ \$ \end{bmatrix} \begin{bmatrix} \star x_1 \\ \star y_1 \star \end{bmatrix} \begin{bmatrix} \star x_{i_{m+2}} \\ y_{i_{m+2}} \star \end{bmatrix} \ldots \begin{bmatrix} \#\$ \\ \$ \end{bmatrix}$

$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} \begin{bmatrix} x_{i_2} \\ y_{i_2} \end{bmatrix} \ldots \begin{bmatrix} x_{i_{m-1}} \\ y_{i_{m-1}} \end{bmatrix}$

$\Rightarrow (1, i_2, i_3, \ldots, i_{m-1})$ is a solution sequence for P

PCP in Coq

```
Definition pcp (t:Type) : Type := list (list t * list t).
Definition mpcp (t:Type) : Type := (list t * list t) * (pcp t).

Definition pcp_instance P :=
  P ≠ [] ∧ ∀ p, p ∈ P → (fst p) ≠ [] ∧ (snd p) ≠ [] .

Definition solution S :=
  concat (map fst S) = concat (map snd S).

Definition pcp_solution P S :=
  S ≠ []
  ∧ S ⊆ P
  ∧ solution S.
```

Reduction MPCP $\leq_M$ PCP in Coq

```
Variable sig': finType.
Inductive sig := # | $ | sigma (s: sig').
```

Theorem mpcp_pcp_reduction:

```
∀ (M: mpcp sig'), mpcp_instance M →
∃ (S': pcp sig'), mpcp_solution M S'
↔ (∃ (S: pcp sig), pcp_solution (mpcp_to_pcp (fst M) M) S).
```

⇒

```
Definition mpcp_to_pcp fcard P : pcp sig :=
  [( ⋆(fst fcard), ⋆(snd fcard)⋆)]
  ++ map (λ p ⇒ (⋆(fst p),(snd p)⋆)) P
  ++ [([#;$], [$])].

Lemma pcp_mpcp_solution (S': pcp sig') fcard:
  solution (fcard::S') → solution (mpcp_to_pcp fcard S').
```

Next Step

$$A_{TM} \leq_m PCP$$

$$A_{TM} \leq_m MPCP \qquad MPCP \leq_m PCP \checkmark$$

References

J. E. Hopcroft, R. Motwani, and J. D. Ullman.
Introduction to Automata Theory, Languages, and Computation.
Pearson, third edition, 2006.

M. Sipser.
Introduction to the Theory of Computation.
Course Technology, second edition, 2006.

Interesting Facts about PCP

- proved undecidable by Emil L. Post with Post canonical systems
- used to prove undecidability of problems for context-free languages
  - G is ambigious
  - $L(G_1) \cap L(G_2) = \emptyset$ for arbitrary CFG G, G_1 and G_2
- restricted versions are decidable[1],
  e.g. $|\Sigma| = 1$ or the number of cards $k \leq 2$
- PCP with 4 cards is undecidable [2]

[1] V. Halava, T. Harju, and M. Hirvensalo; Binary (generalized) post correspondence problem; Theor. Comput. Sci., 276(1-2):183204, 2002.

[2] T. Neary, Undecidability in binary tag systems and the post correspondence problem for four pairs of words, CoRR, abs/1312.6700, 2013

Ambiguity of Context-free Grammars

$PCP \leq_m L_A = \{G \mid \text{CFG G is ambiguous}\}$
Transform a PCP instance P into a context-free grammar G

$$P \text{ has a match} \Leftrightarrow G \text{ is ambiguous}$$

$$P = \left\{ \left[\frac{x_1}{y_1}\right]_{a_1}, \left[\frac{x_2}{y_2}\right]_{a_2}, \ldots, \left[\frac{x_k}{y_k}\right]_{a_k} \right\}$$

$S \to A \mid B$
$A \to x_i A a_i \mid x_i a_i$
$B \to y_i B a_i \mid y_i a_i \qquad \text{for each } \left[\frac{x_i}{y_i}\right] \in P$

Reduction MPCP $\leq_M$ PCP in Coq

$$S = \begin{bmatrix}\star x_1\\ \hline \star y_1 \star\end{bmatrix} \begin{bmatrix}\star x_2\\ \hline y_2 \star\end{bmatrix} \begin{bmatrix}\star x_3\\ \hline y_3 \star\end{bmatrix} \begin{bmatrix}\#\$\\ \hline \$\end{bmatrix} \begin{bmatrix}\star x_1\\ \hline \star y_1 \star\end{bmatrix} \begin{bmatrix}\star x_2\\ \hline y_2 \star\end{bmatrix} \begin{bmatrix}\star x_3\\ \hline y_3 \star\end{bmatrix} \begin{bmatrix}\#\$\\ \hline \$\end{bmatrix}$$

$$S' = \begin{bmatrix}x_1\\ \hline y_1\end{bmatrix} \begin{bmatrix}x_2\\ \hline y_2\end{bmatrix} \begin{bmatrix}x_3\\ \hline y_3\end{bmatrix} \begin{bmatrix}\#\$\\ \hline \$\end{bmatrix} \begin{bmatrix}x_1\\ \hline y_1\end{bmatrix} \begin{bmatrix}x_2\\ \hline y_2\end{bmatrix} \begin{bmatrix}x_3\\ \hline y_3\end{bmatrix} \begin{bmatrix}\#\$\\ \hline \$\end{bmatrix}$$

$\Leftarrow$

Assumption: ∃ (`S: pcp sig`), `pcp_solution (mpcp_to_pcp (fst M) M) S`
Goal: construct solution `S'` of type `pcp sig'` without # and $ symbols

- `S` = [(⋆(`fst fcard`), ⋆(`snd fcard`)⋆)]++`R`
- convert `R:pcp sig` into `R'`: `pcp sig'`
  i.e. delete all # symbols and `([#;$],[$])` cards
- define `S'` = (fst M)::`R'` and show
  `solution S` → `solution (fst M)::R'`
  `(fst M):: R'` ⊆ `M`