Undecidability of Post Correspondence Problem

Meenakshi D'Souza

International Institute of Information Technology
Bangalore

Term II 2022-23

Post Correspondence Problem

- Consider a collection of dominos, each containing two strings, one on each side.
- For example, $\left[\frac{a}{ab}\right]$ is one domino and a collection of dominos looks like $\left\{\left[\frac{b}{ca}\right], \left[\frac{a}{ab}\right], \left[\frac{ca}{a}\right], \left[\frac{abc}{c}\right]\right\}$.
- The problem involves making a list of these dominos (repetitions permitted) so that the string we get by reading off the symbols on the top is the same as the string of symbols on the bottom. Such a list is called a match.
- For example, the following list is a match for the above collection:
  $\left[\frac{a}{ab}\right] \left[\frac{b}{ca}\right] \left[\frac{ca}{a}\right] \left[\frac{a}{ab}\right] \left[\frac{abc}{c}\right]$.
- The Post correspondence problem is to determine whether a collection of dominos has a match.

Post Correspondence Problem

An instance of PCP is a collection P of dominos

$$P = \left\{ \left[\frac{t_1}{b_1} \right], \left[\frac{t_2}{b_2} \right], \ldots, \left[\frac{t_n}{b_n} \right] \right\}$$

and a match is a sequence $i_1, i_2, \ldots, i_n$ where

$$t_{i_1} t_{i_2} \ldots t_{i_n} = b_{i_1} b_{i_2} \ldots b_{i_n}.$$

The problem is to determine whether P has a match.
Let $PCP = \{< P > \mid\ P$ is an instance of PCP with a match $\}$.

PCP is undecidable

Theorem

PCP is undecidable.

PCP is undecidable

Theorem

PCP is undecidable.

- The proof is by reduction from the problem A_{TM}, where $A_{TM} = \{< M, w >| \ M \text{ is a TM and } M \text{ accepts } w\}$.
- We show that from any TM M and input w, we can construct an instance P where a match is an accepting computation history of M on w. If we could determine whether the instance of PCP has a match, we could determine whether M accepts w.
- We choose the dominos in P so that making a match forces a simulation of M to occur. In the match, each domino links a position or positions in one configuration with the corresponding one(s) in the next configuration.

Some modifications to PCP

To make the construction easier, we handle two small technical points.

- If $w = \epsilon$, we use the blank symbol $\sqcup$ in place of w in the construction.
- We also modify PCP to P' which requires that a match starts with the first domino $\left[\frac{t_1}{b_1}\right]$. It is easy to see that this modified version of PCP is also undecidable. We will later see how to do without this modification.

Proof of undecidability of PCP

- Consider a TM R deciding PCP. We construct a TM S deciding A_{TM}.
- Let $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{acc}, q_{rej})$. The TM S constructs an instance of the modified PCP P' that has a match iff M accepts w.
- The construction is described in seven parts, each of which accomplishes a particular aspect of simulating M on w.

Construction of modified PCP

The construction begins in the following manner.

- **Part 1:** Put $\left[\frac{\#}{\# q_0 w_1 w_2 \cdots w_n \#}\right]$ into P' as the first domino $\left[\frac{t_1}{b_1}\right]$. Since we are working with modified PCP, a match must begin with this domino. The bottom string of this domino is the first configuration in an accepting computation history for M on w.

Construction of modified PCP

- **Part 2:** For every $a, b \in \Gamma$ and every $q, r \in Q$ where $q \neq q_{rej}$, if $\delta(q, a) = (r, b, R)$, put $\left[\frac{qa}{br}\right]$ into P'.
- **Part 3:** For every $a, b, c \in \Gamma$ and every $q, r \in Q$ where $q \neq q_{rej}$,
  if $\delta(q, a) = (r, b, L)$, put $\left[\frac{cqa}{rcb}\right]$ into P'.
- **Part 4:** For every $a \in \Gamma$, put $\left[\frac{a}{a}\right]$ into P'.

The dominos of parts 2, 3 and 4 let us extend the match by adding the successive configurations, after the first one.

Construction of modified PCP

Since, in our construction, each configuration is separated by a $\#$ symbol, we need a domino to add the $\#$ symbol.

- **Part 5:** Put $\left[\frac{\#}{\#}\right]$ and $\left[\frac{\#}{\sqcup\#}\right]$ into P'.

The first domino allows us to copy the $\#$ symbol that marks the separation of the configurations.

The second domino allows us to add the blank symbol $\sqcup$ at the end of the configuration to simulate the infinitely blanks to the right that are suppressed when we write the configuration.

Construction of modified PCP

- **Part 6:** For every $a \in \Gamma$, put $\left[\frac{aq_{acc}}{q_{acc}}\right]$ and $\left[\frac{q_{acc}a}{q_{acc}}\right]$ into P'.

This step has the effect of adding “pseudo-steps” of the Turing machine after it has halted, where the head “eats” adjacent symbols until none are left.

Construction of modified PCP

- **Part 7:** Finally, we add the domino $\left[\frac{q_{acc}\#\#}{\#}\right]$ and complete the match.

This concludes the construction of P'.

Conversion of P' to P

Towards converting P' into P, we build the requirement of starting with the first domino directly into the problem so that stating the explicit requirement becomes unnecessary.

Let $u = u_1 u_2 \dots u_n$ be a string of length n. Define $\star u$, $u\star$ and $\star u \star$ to be the three strings

- $\star u = * u_1 * u_2 \dots * u_n$
- $u\star = u_1 * u_2 * \dots u_n *$
- $\star u \star = * u_1 * u_2 * \dots * u_n *$

Conversion of P' to P

To convert P' to P, we do the following. If P' were the collection

$$\left\{ \left[\frac{t_1}{b_1}\right], \left[\frac{t_2}{b_2}\right], \ldots \left[\frac{t_n}{b_n}\right] \right\},$$

we let P be the collection

$$\left\{ \left[\frac{\star t_1}{\star b_1 \star}\right], \left[\frac{\star t_1}{b_1 \star}\right], \left[\frac{\star t_2}{b_2 \star}\right], \ldots, \left[\frac{\star t_n}{b_n \star}\right], \left[\frac{\star \Diamond}{\Diamond}\right] \right\}.$$

Considering P as an instance of the PCP, we see that the only domino that could possibly start a match is the first one, $\left[\frac{\star t_1}{\star b_1 \star}\right]$, because it is the only one where both the top and the bottom start with the same symbol, namely $\star$.

The presence of $\star$ doesn't affect possible matches because they simply interleave with the original symbols. The domino $\left[\frac{\ast \Diamond}{\Diamond}\right]$ is there to allow the top to add the extra $\ast$ at the end of the match.