The Post Correspondence Problem

Prakash Panangaden

23rd March 2021

We are given a *finite set of pairs* of strings over some alphabet Σ:

$$S = \{(\alpha_1, \beta_1),\ (\alpha_2, \beta_2), \ldots (\alpha_k, \beta_k)\}.$$

We represent these strings as dominos; little tiles with one string on top and another string on the bottom as in this example:

$$\left\{ \left[\frac{b}{bbb}\right],\ \left[\frac{babbb}{ba}\right],\ \left[\frac{ba}{a}\right] \right\}$$

We assume we have *as many copies as want* of each *type* of tile; we can represent this solution as a sequence of natural numbers. A description of the tiles is called an instance of the **Post correspondence problem** (PCP). Solving the problem means finding a sequence of tiles (with repetitions if we want) so that when we read the concatenated string on the top of the tiles it is the same as the concatenated string at the bottom of the tiles.

The example of PCP given above has the solution 2113 because we have

$$\begin{array}{cccccc} babbb & b & b & ba & = & babbbbbba \\ 2 & 1 & 1 & 3 & & \\ ba & bbb & bbb & a & = & babbbbbba \end{array}$$

We say that the above instance of PCP *has a solution* or is *solvable*. A PCP *may* have many nontrivial solutions or **none**. For example, the following

$$\left\{ \left[\frac{ab}{abb}\right],\ \left[\frac{b}{ba}\right],\ \left[\frac{a}{bb}\right] \right\}$$

has no solution because the strings on top are all strictly shorter than the strings below.

Theorem 1 (Post)**.** It is undecidable whether a given Post system has a solution.

Proof . Given a Turing machine M and a word w, we give an effective construction of a PCP such that, the solution to the PCP given a valid computation for M run with w as input. This VALCOMPS$(M, w) = \emptyset$ if and only if the PCP instance has no solution.

We design the domino tiles so that in a match each tile links a configuration of M with the next configuration. We use # as a special separator symbol. As usual we represent configurations as $w_1 q w_2$ where w_1 is the word to the left of the read head of M, q is the state of M and w_2 is the word that starts with the symbol on the cell that the read head is looking at and continues to the right on the tape.

We make a *temporary* restriction on PCP: we insist that there is a special designated tile called the *start* tile and every solution is required to start with this tile. Later we will eliminate this restriction.

We insist that the starting tile is

$$\left[\frac{\#}{\#q_0 w_0 \dots w_n}\right]$$

where q_0 is the start state of M and the word w consists of the symbols $w_0 \dots w_n$.

For every transition of the form $\delta(q, a) = (r, b, R)$ with q not equal to the reject state we make a tile of the form

$$\left[\frac{qa}{br}\right]$$

For every transition of the form $\delta(q, a) = (r, b, L)$ with q not equal to the reject state we make a tile of the form

$$\left[\frac{cqa}{rcb}\right]$$

for every $c \in \Sigma$.

For every $a \in \Gamma$ we create a tile of the form

$$\left[\frac{a}{a}\right].$$

We add $\left[\frac{\#}{\#}\right]$ and $\left[\frac{\#}{\sqcup\#}\right]$.

Finally we add three more tiles

$$\left[\frac{aq_a}{q_a}\right], \left[\frac{q_a a}{q_a}\right], \left[\frac{q_a\#\#}{\#}\right].$$

The tiles force one to construct a valid computation. Suppose that $\Gamma = \{0, 1, 2, \sqcup\}$ and $w = 0100$ and $\delta(q_0, 0) = (q_7, 2, R)$. We start with the start tile

$$\left[\frac{\#}{\#q_0 0100\#}\right]$$

and to continue the match we have to use a tile that starts with $q_0 0$ on top so doing this and then adding tiles to complete the extra letters we get

$$\left[\frac{\#}{\#q_0 0100\#}\right]\left[\frac{q_0 0}{2q_7}\right]\left[\frac{1}{1}\right]\left[\frac{0}{0}\right]\left[\frac{0}{0}\right]\left[\frac{\#}{\#}\right]$$

as you can see the top is now matched but the bottom has extra symbols that *exactly correspond to the next configuration.* The last few tiles are used to finish up the computation as follows.

Suppose that M enters an accept state so the tiles are now aligned as follows (for example):

$$\left[\frac{\#}{\#21q_a 02\#}\right].$$

We use the tiles

$$\left[\frac{aq_a}{q_a}\right], \left[\frac{q_a a}{q_a}\right]$$

to catch up and to finish the computation we use

$$\left[\frac{q_a\#\#}{\#}\right]$$

as follows

$$\left[\frac{\#}{\#q_a\#}\right]\left[\frac{q_a\#\#}{\#}\right].$$

The last detail is generalizing the modified version of the Post correspondence problem to the original version. We suppose that $*$ and $\diamond$ are symbols that we have never used before. If $u = u_0 \dots u_n$ is a string, we define $*u := *u_0 * \dots * u_n$ and $u* := u_0 * \dots u_n*$ and $*u* = *u_0 * \dots * u_n*$. For

every tile $\left[\frac{t}{b}\right]$ in our original set of tiles we add $\left[\frac{*t}{b*}\right]$ and for the start tile $\left[\frac{t_0}{b_0}\right]$ we add $\left[\frac{*t_0}{*b_0*}\right]$; we have to start with this tile otherwise no match is possible. Finally, to end we add the tile $\left[\frac{*\diamond}{\diamond}\right]$.

So we have *effectively constructed* a set of tiles with the property that there is a solution if and only if M accepts w. This is thus, even a mapping reduction. If PCP were solvable we could solve the acceptance problem for Turing machines which is known to be undecidable. ∎