Pretty printing—a problem in your daily life

We would like to print a given document nicely, e.g.,

- right ends of lines are justified,
- spaces between words (or letters) are as even as possible,
- the number of words broken into two lines (by using "-") is as small as possible.

Applications: newspapers, books, word processors, TEX, LATEX, etc.

Layout examples:

C'erano una volta tre porcellini fratelli che vivevano con i loro genitori. I tre porcellini mangiavano bevevano e si divertivano in tutti i giorni di sole, mentre in quelli di pioggia si riposavano e con il passare del tempo finirono per crescere ben pasciuti.

C'erano una volta tre porcellini fratelli che vivevano con i loro genitori. I tre porcellini mangiavano bevevano e si divertivano in tutti i giorni di sole, mentre in quelli di pioggia si riposavano e con il passare del tempo finirono per crescere ben pasciuti.

C'erano una volta tre porcellini fratelli che vivevano con i loro genitori. I tre porcellini mangiavano bevevano e si divertivano in tutti i giorni di sole, mentre in quelli di pioggia si riposavano e con il passare del tempo finirono per crescere ben pasciuti.

(↖) not right-justified

(↑) right-justified;

line-breaks: the same as (↖)

(←) typeset by LaTeX

Below we assume for simplicity that it is not allowed to divide a word.

The algorithm below can easily be generalized to the case where line-breaking in a word is allowed and spacing between characters (even in a word) can vary. In LaTeX, both of these are considered!

Input: The width l_i of each word w_i $(i = 1, 2, \ldots, n)$,
an ideal space width s, and a text width L.

When words from w_i to w_j $(1 \leq i \leq j \leq n)$ are written in a line, the width of each space between words is

$$s(i,j) = \frac{L - \sum_{k=i}^{j} l_k}{\max\{j-i, 1\}},$$

which incurs a cost

$$c(i,j) = \begin{cases} +\infty, & \text{if } s(i,j) \leq 0 \\ (s - s(i,j))^2, & \text{if } s(i,j) > 0 \text{ and } j < n \\ \max\{s - s(i,j), 0\}^2, & \text{if } s(i,j) > 0 \text{ and } j = n. \end{cases}$$

Output: A layout (points to break lines) that minimizes the total cost.

DP for pretty printing

$f^*(i)$: the cost of the best layout for words $w_i, w_{i+1}, \ldots, w_n$.

The recurrence formula for $f^*(i)$:

$$f^*(i) = \begin{cases} 0, & i = n+1 \\ \min_{i \le j \le n} \{c(i,j) + f^*(j+1)\}, & i \le n. \end{cases}$$

Computation time of the DP:

- If $c(i,j)$ is computed from scratch $\Longrightarrow$ $\mathrm{O}(n^3)$ time in total.
- Compute $c(i,j)$ in $\mathrm{O}(1)$ time using $\sum_{k=i}^{j} l_k = \sum_{k=i}^{j-1} l_k + l_j$

  $\Longrightarrow$ $\mathrm{O}(n^2)$ time in total.

Note: For cost function $c(i,j)$, you can use different functions, e.g., cubic power, as long as its computation is easy.