Shortest common supersequence problem

From Wikipedia, the free encyclopedia

In computer science, the **shortest common supersequence** of two sequences X and Y is the shortest sequence which has X and Y as subsequences. This is a problem closely related to the longest common subsequence problem. Given two sequences X = < $x_1,...,x_m$ > and Y = < $y_1,...,y_n$ >, a sequence U = < $u_1,...,u_k$ > is a common supersequence of X and Y if items can be removed from U to produce X or Y.

A shortest common supersequence (SCS) is a common supersequence of minimal length. In the shortest common supersequence problem, the two sequences X and Y are given and the task is to find a shortest possible common supersequence of these sequences. In general, an SCS is not unique.

For two input sequences, an SCS can be formed from a longest common subsequence (LCS) easily. For example, if X$[1..m] = abcbdab$ and Y$[1..n] = bdcaba$, the lcs is Z$[1..r] = bcba$. By inserting the non-lcs symbols while preserving the symbol order, we get the SCS: U $[1..t] = abdcabdab$.

It is quite clear that $r + t = m + n$ for two input sequences. However, for three or more input sequences this does not hold. Note also, that the lcs and the SCS problems are not dual problems.

For the more general problem of finding a string, S which is a supersequence of a set of strings $S_1,S_2,...,S_l$, the problem is NP-Complete .[1] Also, good approximations can be found for the average case but not for the worst case.[2][3]

References

1. Kari-Jouko Räihä, Esko Ukkonen (1981). "The shortest common supersequence problem over binary alphabet is NP-complete" (http://www.sciencedirect.com/science/article/pii/030439758190075X). *Theoretical Computer Science*. **16** (2): 187 – 198. doi:10.1016/0304-3975(81)90075-x (https://doi.org/10.1016%2F0304-3975%2881%2990075-x).
2. Tao Jiang and Ming Li (1994). "On the Approximation of Shortest Common Supersequences and Longest Common Subsequences" (http://epubs.siam.org/doi/abs/10.1137/S009753979223842X). *SIAM Journal on Computing*. **24** (5): 1122 – 1139. doi:10.1137/s009753979223842x (https://doi.org/10.1137%2Fs009753979223842x).
3. Marek Karpinski and Richard Schmied (2013). "On Improved Inapproximability Results for the Shortest Superstring and Related Problems" (http://crpit.com/abstracts/CRPITV141Karpinski.html). *Proceedings of 19th CATS CRPIT*. **141**: 27 – 36.

- Garey, Michael R.; Johnson, David S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman. p. 228 A4.2: SR8. ISBN 0-7167-1045-5. Zbl 0411.68039 (https://zbmath.org/?format=complete&q=an:0411.68039).
- Szpankowski, Wojciech (2001). *Average case analysis of algorithms on sequences*. Wiley-Interscience Series in Discrete Mathematics and Optimization. With a foreword by Philippe Flajolet. Chichester: Wiley. ISBN 0-471-24063-X. Zbl 0968.68205 (https://zbmath.org/?format=complete&q=an:0968.68205).

External links

- Dictionary of Algorithms and Data Structures: shortest common supersequence (http://nist.gov/dads/HTML/shortestCommonSuperseq.html)

Categories: Problems on strings | Combinatorics | Formal languages | Dynamic programming