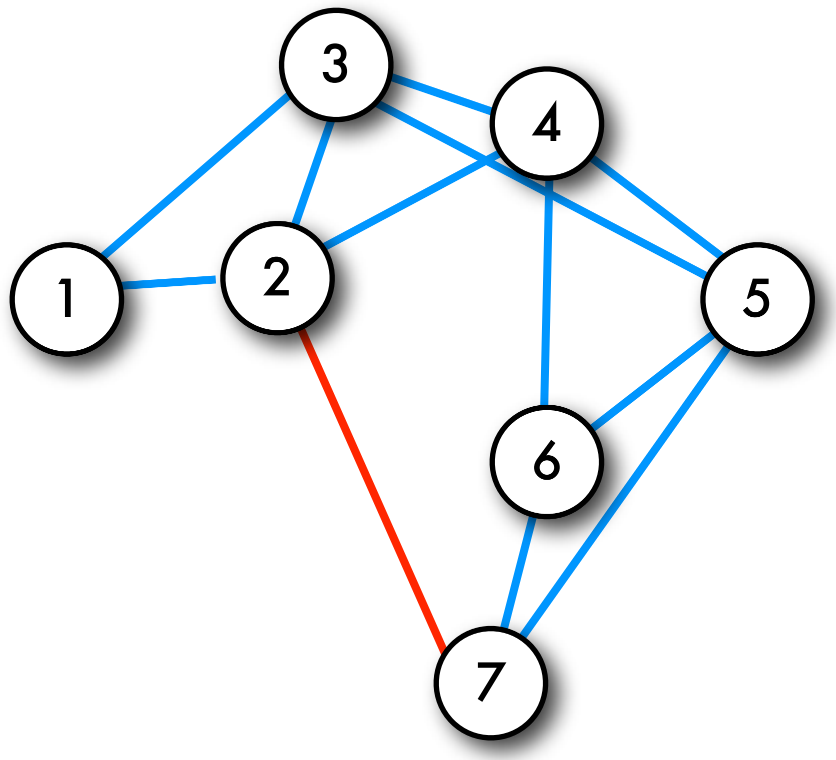
The Interval Ordering Problem

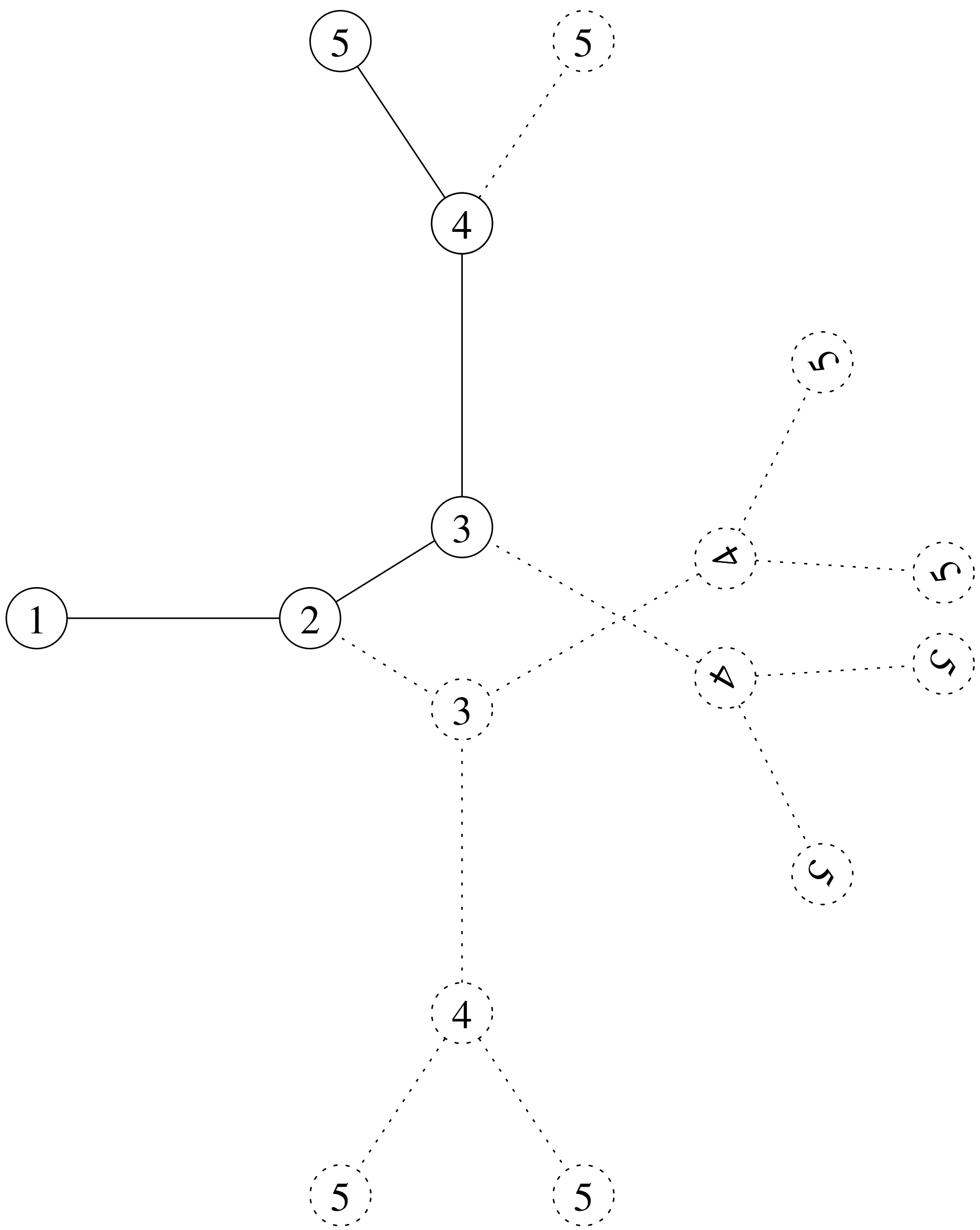
with Maurice Queyranne, Frits Spieksma, Fabrice Talla Nobibon, Gerhard Woeginger

A motivation



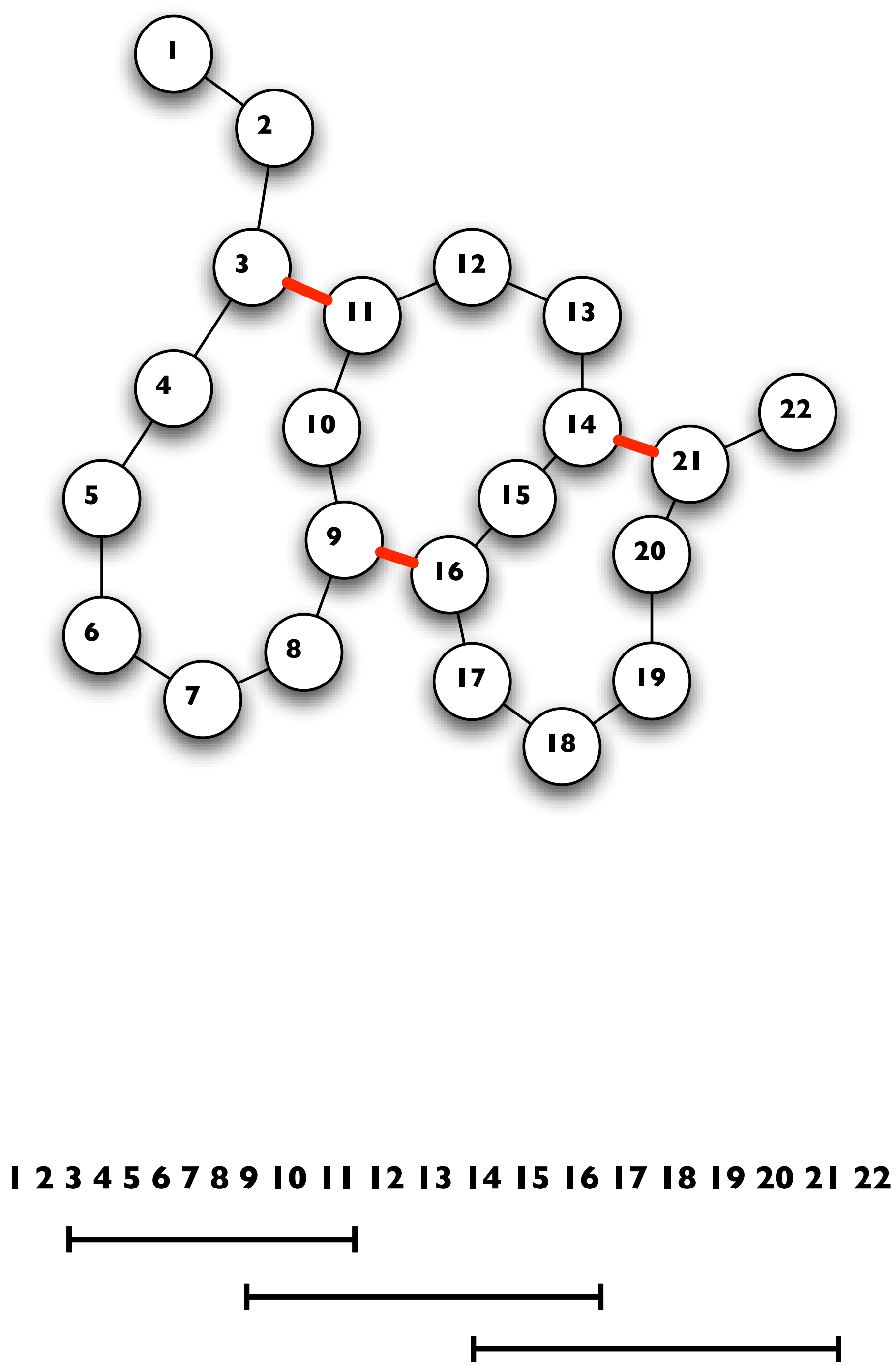
- A molecule consisting of atoms 1,..,n in unknown positions
- we are given the distances between all atom pairs (i,i+1) and (i,i+2)
- and the distances between some other atom pairs

a combinatorial structure

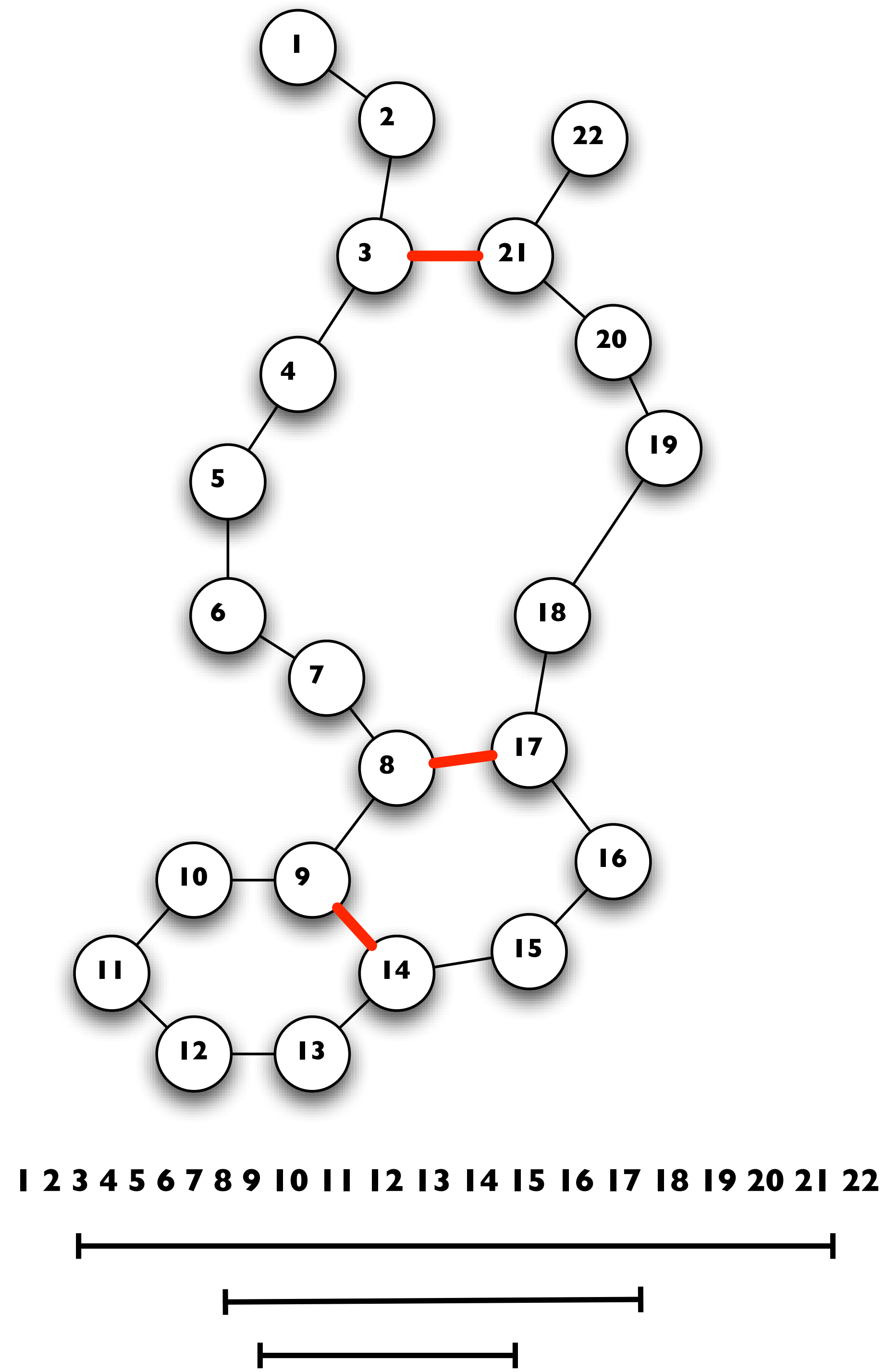


- with the distances (i-1,i) and (i-2,i), there are 2 possible positions for atom i relative to atoms i-1,i-2.
- A binary string describes all valid embeddings
- the other distances are constraints on substrings

special interval families



agreeable intervals

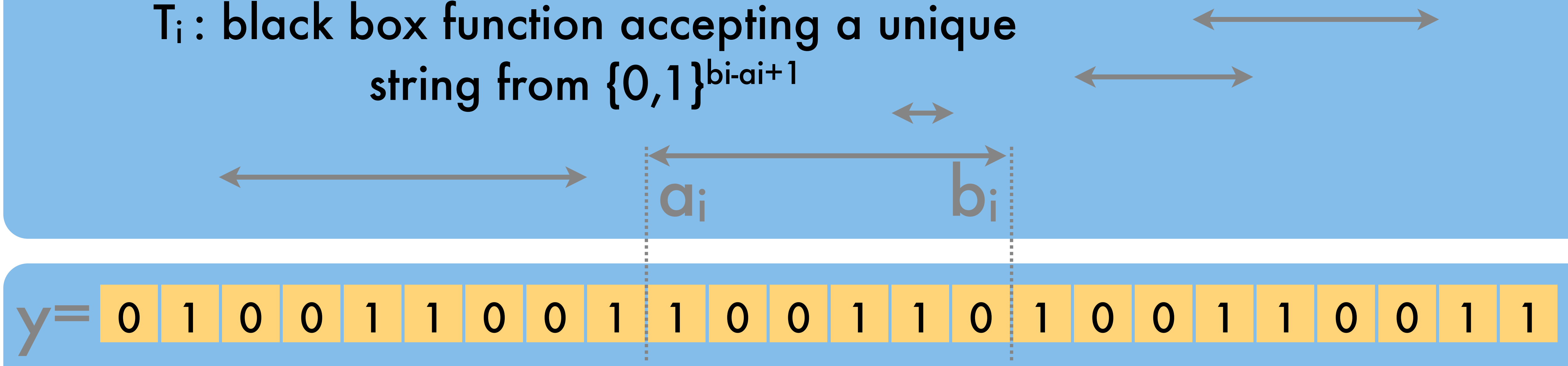


laminar intervals

The Bit String Reconstruction Problem

input $(a_1, b_1, T_1) \ldots (a_n, b_n, T_n)$

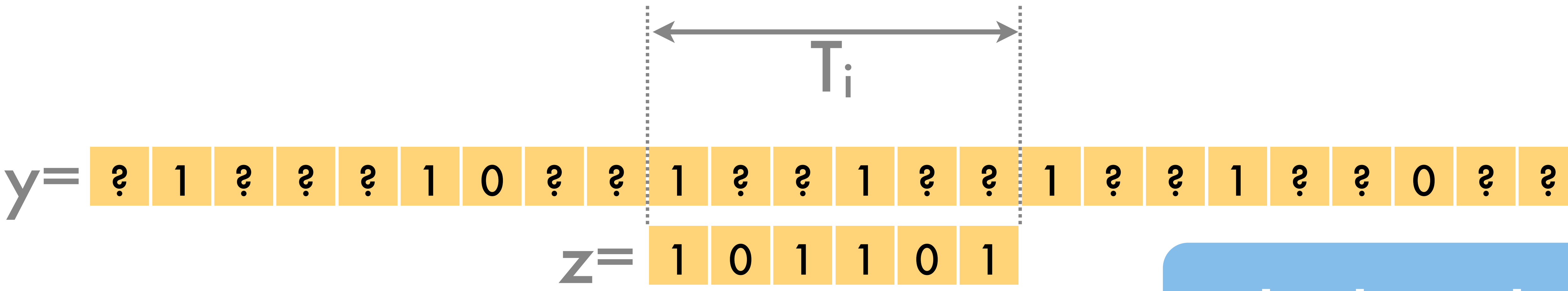
T_i : black box function accepting a unique string from $\{0,1\}^{b_i - a_i + 1}$



such that $\forall i$: T_i accepts the substring $y[a_i \ldots b_i]$

output (or announce inconsistence)

The Brute Force Search



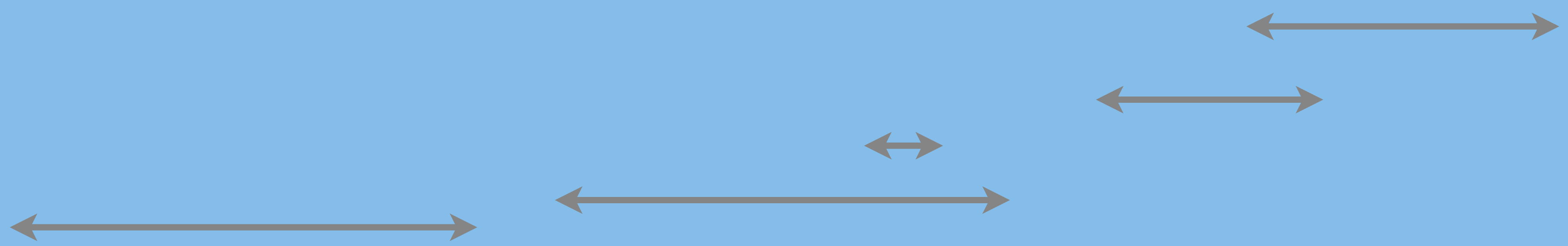
- $y \in \{?,0,1\}^m$, start with $y=?^m$
- for all constraints (a_i,b_i,T_i) in some order
  - $w=y[a_i,b_i]$, k=number of ? in w
  - try all 2^k substitutions of ? by 0 or 1 in w
  - until we find a z accepted by Ti
  - replace in y the portion $y[a_i,b_i]$ by z

which order on those constraints leads to the smallest running time ?

The Interval Ordering Problem

fixed a function f

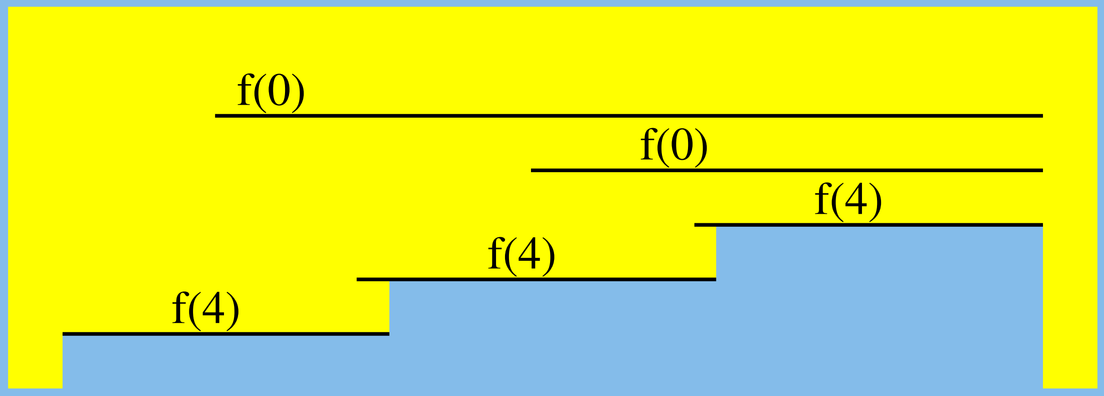
input n intervals $I_1,\dots,I_n$



output an order on these intervals

minimizing

$$\sum_k f(I_k \setminus (I_1 \cup \dots \cup I_{k-1}))$$

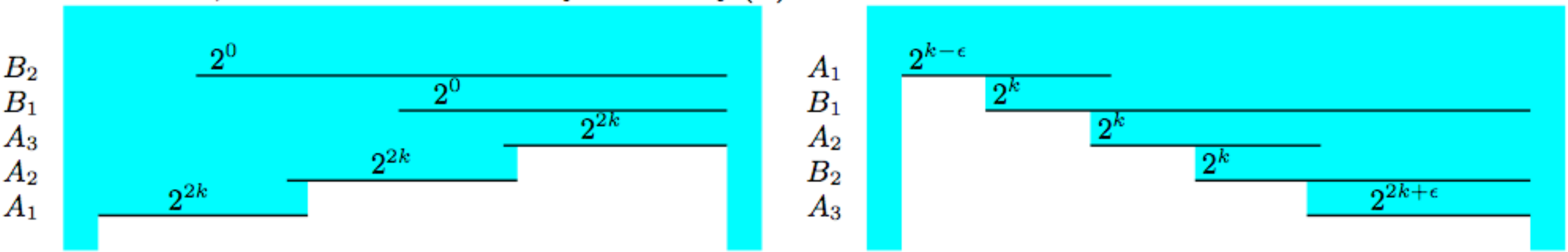


greedy ordering with respect to exposed length

How bad is the greedy ordering?

for $f(x)=2^x$ ...

Example 2. *Consider a family of instances, where each instance consists of $2k-1$ intervals: $A_1 = [0, 2k)$, $A_2 = [2k-\epsilon, 4k)$, $A_3 = [4k-\epsilon, 6k), \ldots$, $A_k = [2k(k-1)-\epsilon, 2k^2)$, $B_1 = [k-\epsilon, 2k^2)$, $B_2 = [3k-\epsilon, 2k^2)$, $B_3 = [5k-\epsilon, 2k^2), \ldots$, $B_{k-1} = [2k^2-3k-\epsilon, 2k^2)$, for some constants k, $\epsilon > 0$ with the cost function $f(x) = 2^x$.*



greedy ordering with respect to exposed length

optimal ordering

A greedy sequence is $(A_1, A_2, \ldots, A_{k-1}, A_k, B_{k-1}, B_{k-2}, \ldots, B_1)$ and achieves a cost of $k2^{2k} + k - 1$, whereas the optimal solution is $(A_k, B_{k-1}, A_{k-1}, B_{k-2}, \ldots, A_2, B_1, A_1)$ and has the cost of $2^{2k+\epsilon} + (2k-3)2^k + 2^{k-\epsilon}$. The ratio between both costs can be made arbitrarily large, by choosing appropriate k and small $\epsilon > 0$.

... arbitrary bad!

What do we know?

+

- f arbitrary, (I) agreeable : dynamic programming in $O(n^3)$
- f continuous, convex, (I) agreeable : dynamic pro. in $O(n^2)$
- f(x)-f(0) sub-additive, (I) laminar : inner to outer in O(nlogn)
- variant: minimize $\max_k$ f(exposed part of I_k) : greedy in $O(n^2)$
- $f(x)=2^x$, (I) arbitrary : open

−

- f arbitrary, I arbitrary cannot be constant approximated (unless P=NP)