Popis

Budeme vycházet z myšlenky základní verze, ve které jsme z řádků udělali jednu partitu a ze sloupců druhou. Nyní to však nebudou celé řádky a sloupce, ale řádkové a sloupcové segmenty. Formálně tedy každý řádek rozdělíme na segmenty (maximální úseky bez děr) a ty vložíme do množiny A. Obdobně se sloupci, jejichž segmenty vložíme do množiny B. Hrany mezi $a \in A$ a $b \in B$ existují tehdy, pokud se segmenty kříží. Na výsledném bipartitním grafu stačí nalézt maximální párování (hrana značí, že se na křížícím místě položí věž).

Důkaz

Nalezené rozestavení je validní, neboť pokud by nebylo, tak jsme umístili věže tak, že se ohrožují, tedy nachází se ve stejném segmentu. To by však znamenalo, že z vrcholu (reprezentující daný segment) vedou alespoň dvě hrany, což je spor s definicí párování. Naopak každé validní rozestavení má ekvivalentní párování v naší reprezentaci. Po obměně: jestliže není něco v naší reprezentaci párování, tak to není validní rozestavení. Opět platí argument s více hranami z jednoho vrcholu.

Algoritmus a složitost

Segmentů může vzniknout až $O(n^2)$, hran též (n je šířka/výška desky). Pro hledání maximálního párování použijeme nějaký algoritmus největšího toku (zapojíme jednu partitu do zdroje, druhou do stoku a všemu dáme kapacitu 1). Nejefektivnější umí řešit až v $O(VE)$, tedy celkem $O(n^4)$.

PS: Až po dopsání jsem si všiml, že se jedná o šachovnici (8×8). Tedy je vše konstanta a náš program seběhne v $O(1)$.