

EEPROM

Zadatak rješavamo metodom dinamičkog programiranja. Podatke zapisujemo u memorijski modul po recima, a unutar redaka po stupcima. U svakom trenutku jedini relevantni podaci su (tzv. parametri stanja podproblema):

- r, s – redak i stupac polja u koji upisujemo sljedeći bit
- hor – oznaka jesmo li polje neposredno lijevo polju (r, s) mijenjali horizontalnim potezom
- $mask$ – bitmaska koja za svaki stupac označava jesmo li zadnje obrađeno polje u tom stupcu mijenjali vertikalnim potezom

Iz gornjih podataka, moguće je izvesti rekursivnu relaciju koja daje optimalno rješenje danog podproblema. Kako u relaciji koristimo samo podatke iz prethodnog retka tablice, moguće je pamtit i samo posljednja dva retka, te na taj način smanjiti potrošnju memorije.

TORNJEVI

Za svaki toranj zapravo možemo odvojeno odrediti puca li lijevo ili desno, te puca li gore ili dolje. Ovo možemo modelirati dvjema logičkim varijablama po svakom tornju. Za toranj i :

- Ako toranj puca lijevo, onda je varijabla L_i istina
- Ako toranj puca gore, onda je varijabla G_i istina
- Ako toranj puca desno, onda je varijabla L_i laž, tj. izraz $\neg L_i$ ("ne L_i ") je istina
- Ako toranj puca dolje, onda je varijabla G_i laž

Vrijednosti nekih varijabli su unaprijed određene: npr. ako se dva tornja nalaze u istom redu bez dvoraca između, tad lijevi toranj ne smije pucati desno, a desni ne smije pucati lijevo.

Dodatne uvjete na vrijednosti varijabli postavljaju položaji napadača. Svakog napadača mora pogoditi barem jedan toranj. Međutim, nije moguće da postoji više od dva tornja kandidata za gađanje jednog napadača, jer bi to značilo da su dva od ta tri tornja u istom redu ili stupcu, pa bi jedan toranj uništio drugoga (nakon što pogodi napadača). Stoga svaki napadač ili:

- Jednoznačno određuje vrijednost neke varijable (ako je samo jedan toranj kandidat), ili
- Kaže "mora me pogoditi jedan od ova dva tornja", postavljajući logički uvjet oblika npr.
 $L_A \vee \neg G_B$ ("mora me pogoditi toranj A pucajući lijevo ili toranj B pucajući dolje").

Potrebno je odabrati vrijednosti varijabli L_i i G_i tako da svi uvjeti budu zadovoljeni. Prikladno, ovaj problem zove se problem zadovoljavanja (engl. satisfiability). U općenitom slučaju nije efikasno rješiv, ali jest kad svaki uvjet sadrži najviše dvije varijable, kao u ovom zadatku (2-SAT). Više algoritama rješava ovaj problem. Jedan je: za svaku varijablu (koja još nije određena), pretpostavi da je laž pa provjeri (rekursivno) dovodi li do kontradikcije – ako da, onda mora biti istina. Također, pretpostavljanje vrijednosti jedne varijable implicira vrijednosti nekih drugih varijabli, na što treba paziti.