

EXPLICACIO DELS PRINCIPALS ALGORISMES:

Solver:

Aquest és un solucionador que utilitza la tecnica recursiva del backtracking. Com a precondicions tenim:

- 1.El tauler d'entrada es un tauler de tipus Hidato. Per tant, el taulell té com a mínim col·locades les posicions de principi i final.
- 2.La primera crida de l'algorisme es fa sempre desde el punt de partida, el numero 1.

L'algorisme funciona de la següent manera: Parteix del primer nombre i acte seguit de forma ciclica repassa les caselles veïnes. L'algorisme es repetira de forma recursiva sobre aquelles caselles que son buides o bé que contenen el nombre de la casella actual+1. Amb aquelles que son buides fa la crida recursiva assignant'ls-hi el nombre actual de la casella + 1. L'algorisme finalment arriba al numero final habent completat tot el tauler. Al arribar a aquest punt assigna la solució trobada actualment al tauler i fa marxa enrere tornant a posar les caselles buides aquelles que ho eren desde un principi.

Si l'algorisme no troba una solució en menys del temps que li marca el paràmetre Temporitzador, s'assumeix que el tauler no té solució.

Generador:

Aquest és l'algorisme encarregat de generar un tauler de tipus Hidato de manera aleatoria. Com a precondicions tenim que:

- 1.El tauler d'entrada pot tenir forats i només té col·locat el primer numero. (Ha estat col·locat previament de manera aleatoria)
- 2.La primera crida de l'algorisme es fa sempre desde el punt de partida, el numero 1.

L'algorisme funciona de la següent manera: Parteix del primer nombre i esculleix una casella veïna de forma aleatoria que pugui ocupar (que no hi hagi un altre nombre o que no sigui un forat). Acte seguit li assigna el valor de la casella actual+1 i fa una crida recursiva partint d'aquella casella. El tauler acaba completament ple, per tant és una solució possible del tauler.

Unica solució:

L'algorisme es idèntic al solucionador, la diferencia principal és que al arribar al últim nombre retorna un 1, per tant, si l'algorisme arriba per un camí diferent al últim nombre del tauler aquest valor de retorn s'incrementarà. Anàlogament, si l'algorisme no troba una solució en menys del temps que li marca el paràmetre Temporitzador, s'assumeix que el tauler no té solució.

VALORS ESTUDIATS:

L'algorisme de generació d'Hidatos té un fort component d'aleatorietat. El temps d'espera de generació d'un hidato aleatori també està condicionat per la forma:

- Sense forma: Suposarem que un hidato amb un nombre $(n*n)/2$ de caselles buides i negres respectivament. Trio aquest valor ja que l'Hidato resultant és jugable i amb un cert nivell de dificultat. A partir de $n > 9$ l'algorisme ja triga més del necessari.

- Esfera: Aquesta forma té un elevat nombre de forats i no es gaire recomanable afegir-hi gaires de nous. Normalment es generen una mica menys de $(n*n)/2$ forats. Aquest tipus d'hidatos comencen a trigar més del desitjat a partir de $n > 11$. En aquest cas cal ser molt curós a l'hora de triar el nombre de caselles buides ja que aquest mapa conté un elevat nombre de caselles negres.

- Diagonal: Aquest tipus d'Hidatos són realment difícils de solucionar. Sense afegir cap forat extra les crides a partir de $n > 6$ ja comencen a trigar més del desitjat.

Per altra banda, l'algorisme que s'encarrega de solucionar un Hidato (i analogament el de solució única) funciona millor en els casos en que hi ha un baix nombre de caselles buides. Ja que l'algorisme fa una crida recursiva per cada buit i s'incrementen exponencialment.