① N-Puzzle or sliding puzzle

a) Breadth first search algorithm.

As we know that in a sliding game puzzle of 'r' rows and 'c' columns the total possible states can be $= (r \times c)!$

in which $(r \times c)!/2$ are solvable and rest are not solvable.

let, it is a 3×3

then the possible states are $= 9! = 362880$

and the no. of edges = states × avg degree

avgerage degree = 4 with 2 neighbour
+ 4 with 3 neighbour
+ 1 with 4 neighbour

$$= 2 \times \frac{4}{9} + 3 \times \frac{4}{9} + 4 \times \frac{1}{9} = 2.67$$

∴ no. of edges $= 362880 \times 2.67$

Therefore the running time $= O(V+E)$
no. of steps $= 362880 * 3.67$

let it takes 3 sec to iterating $362880 * 3.67$ steps

then for 4×4.

possible states $= 16!$

edges average $= 3$

∴ No. of edges $= 3 \times 16!$

running time $= O(V+E)$
steps $= 16 \times 15 \times 14 \times 13 \times 12 \times 11 \times 10$
$*$ (steps for 3×3)

The estimated time is approx 3.65 years
So the runtime of this algo is largly exponential and it will take more time for large n.

and the memory required would be

$= O(V)$; where V is no of vertices

$= O((r \times c)!)$

b) Depth first search algorithm.

worst case running time $= O(V+E)$ } Same as Bfs

space $= O(V)$ }

~~and~~ but dfs is slower than Bfs

because dfs will ignore many steps.

c) A* algorithm.

for a 3×3 puzzle.

the average branching is 2.67

and no. of possible states $= (3 \times 3)!$

$\therefore (2.67)^d = 9!$

$d \approx 13$

Therefore the approx ~~x~~ no of steps for the 3×3 is 13, and for 4×4 the approx no of steps is 27

~~As a general, for~~