

We had mainly focused on Path-finding Problem (<http://en.wikipedia.org/wiki/Pathfinding>) and explored it to understand the current status of it's research.

Path-finding Problem is to find optimal route between 2 points (2 states in context of AI).

=> Breadth First Search is the starting point for the problem. It's the most basic algorithm but has it's limitation.

Key Point - Depth of a node is used as cost function.

Link - <http://intelligence.worldofcomputing.net/ai-search/breadth-first-search.html#.UcuedM7Tbh8>

Limitations - Memory requirements and if destination is very far from source then it will take a lot of time.

=> Uniform Cost Search a.k.a Dijkstra's Algorithm. Modified BFS as edges have different cost.

Key Point - Cost from the source node is used as cost function called $g(n)$.

Link - <http://intelligence.worldofcomputing.net/ai-search/uniform-cost-search.html#.UcucKc7Tbh8>

Limitations - Same as BFS.

=> Pure Heuristic Search

Key Point - uses estimated value from a node to destination as it's cost function $h(n)$ and does ignore cost of the path so far. It expands and brings in the node.

Link - <http://intelligence.worldofcomputing.net/ai-search/uniform-cost-search.html#.UcucKc7Tbh8>

Limitation - Does not find optimal solutions.

BFS, Uniform Cost Search and Pure Heuristic Search are all special cases of a more general algorithm called BEST FIRST SEARCH with different cost functions.

=> Depth First Search and Iterative Deepening DFS(IDDFS).

Key Point - Removes limitation of memory on all BEST FIRST Algorithms and used in Iterative Deepening in A Star Algorithm even.

Link - http://en.wikipedia.org/wiki/Iterative_deepening_depth-first_search

=> A* Star Algorithm

Key Point - Combines Uniform Cost Search ($g(n)$) and Pure Heuristic Search ($h(n)$) to give $f(n) = g(n) + h(n)$

Link - <http://intelligence.worldofcomputing.net/ai-search/a-star-algorithm.html#.Ucuelc7Tbh8>

For detailed and clear understanding - <http://www.raywenderlich.com/4946/introduction-to-a-pathfinding>

Limitations - Same as BFS.

=> Iterative Deepening A* Star Algorithm

Name is enough for understanding after reading IDDFS.

Link - <http://intelligence.worldofcomputing.net/ai-search/iterative-deepening-a-star.html>

=> Real Time A* Star Algorithm (RTA*)

Key Point - Uses Previous Stored $h(n)$ from visited nodes else heuristic evaluator is called and it will estimate a heuristic value.

Link - <http://intelligence.worldofcomputing.net/ai-search/real-time-a-star.html#.Ucudz7Tbh8>

=> Learning Real Time A* Star Algorithm (LRTA*)

Key Point - Performance increases with time and heuristic functions changes with time and optimal values will be used for next problem instance.

Link - <http://intelligence.worldofcomputing.net/ai-search/learning-real-time-a-star.html#.Ucudys7Tbh8>

Conclusions:

A nice link to visualize these algorithms is here - <http://qiao.github.io/PathFinding.js/visual/>