

نحوه ی مدل کردن مسئله

state به صورتی تعریف شده که وضعیت سر و بدن مار و دانه‌های باقی مانده را نشان دهد.
با تعریف کردن state به این صورت، هر گام مار برای طی کردن مسیرها می‌تواند از یک دیگر تفکیک شود.

initial_state

با توجه به یک بود طول مار در شروع، body ندارد.
همه‌ی دانه‌های خوانده شده از فایل روی صفحه‌بازی بازی موجود هستند.

goal state

همه‌ی دانه‌های بازی خورده شده‌اند.
سر و بدن مار می‌تواند هر وضعیتی بسته به مسیری که رفته داشته باشد.

action

راست، چپ، بالا، و پایین رفتن مار هر کدام می‌تواند اکشن مار در هر یک از گام‌ها برای رسیدن به دانه‌ها باشد.

heuristicها

heuristic اول: فاصله خطی سر مار و دونه

از آن جایی که کوتاه ترین فاصله بین دو نقطه، فاصله خطی آن‌هاست؛ بنابراین این heuristic هم admissible و هم consistent می‌باشد.

heuristic دوم: کوتاه‌ترین فاصله ی L شکل بین سر مار و دونه

این heuristic به واقعیت مسئله ی ما نزدیک تر هم هست؛ چراکه مار نمی‌تواند اریب حرکت کند و فقط گام‌های در یکی از جهت‌های افقی و یا عمودی ممکن می‌تواند داشته باشد.

الگوریتم‌های پیاده سازی شده و تفاوت‌ها و مزیت‌های الگوریتم‌ها نسبت به یکدیگر

BFS

BFS با شروع از نود root، تمام نودهای همسایه در عمق فعلی را به جای رفتن به نودهایی در عمق بعدی پیمایش می‌کند.

IDS

IDS، IDDFS را برای عمق‌های مختلف با شروع از مقدار اولیه صدا می‌زند. در هر بار صدا زدن، DFS از جلو رفتن از بیشینه عمق داده شده منع می‌شود. بنابراین انگار ما DFS را با روش BFS انجام می‌دهیم.

A*

این الگوریتم مسیری که $f(n) = g(n) + h(n)$ کمینه‌ای دارد را در هر بار انتخاب می‌کند؛ که در آن $g(n)$ هزینه مسیر نود شروع تا نود n و $h(n)$ تابع هیوریستیکی است که هزینه بهینه‌ترین مسیر از n به goal را تخمین می‌زند.

BFS vs IDS

When to prefer IDS?

- When a lower space complexity is required and somewhat lower time complexity is acceptable

When to prefer BFS?

- If the solution is in the upper part of the tree as it is less costly to generate the nodes once

BFS vs A*

When to prefer A*?

- If memory space is limited
- If the tree has a high branching factor
- If the tree is dense
- Although the complexity of queue is slightly better than that of priority queue, A*'s time complexity is usually better than BFS's time complexity with a good enough heuristic

When to prefer BFS?

- If the tree has a low branching factor
- If the tree is dense
- If the heuristic is poor
- If the heuristic is not admissible and optimality is required

IDS vs A*

When to prefer IDS?

- When the A* algorithm has a poor heuristic
- When optimality is required and the A* heuristic is not admissible
- If there are limitations to the amount of memory available

When to prefer A*?

- In general uninformed search is slower than informed search, therefore for a good enough heuristic, A* should be preferred