فضای حالت برای مسئله پارچ‌های آب

برای این مسئله، یک جدول جامع برای تعریف فضای حالت ارائه می‌دهیم

مؤلفه	توضیحات
States	هر حالت به صورت (x, y) نمایش داده می‌شود که x مقدار آب در پارچ 4 لیتری y مقدار آب در پارچ 3 لیتری **محدوده** $0 \le x \le 4$ و $0 \le y \le 3$
Initial State	**هر دو پارچ خالی هستند** - (0, 0)
Goal State	**پارچ 4 لیتری باید دقیقاً 2 لیتر آب داشته باشد** (2, y)
Actions	**پر کردن پارچ 4 لیتری**: (x, y) → (4, y) **پر کردن پارچ 3 لیتری**: (x, y) → (x, 3) **خالی کردن پارچ 4 لیتری**: (x, y) → (0, y) **خالی کردن پارچ 3 لیتری**: (x, y) → (x, 0) **ریختن از 4 لیتری به 3 لیتری**: (x, y) → (max(0, x-(3-y)), min(3, x+y)) **ریختن از 3 لیتری به 4 لیتری**: (x, y) → (min(4, x+y), max(0, y-(4-x)))
Transition Model	**اعمال حالت فعلی را به حالت جدید تبدیل می‌کنند با توجه به قوانین فیزیکی ظرفیت پارچه‌ها**
Goal Test	**(پارچ 4 لیتری دقیقاً 2 لیتر دارد)** بررسی x == 2
Path Cost	**هر عمل هزینه 1 واحد دارد (تعداد گام‌ها برای رسیدن به هدف)**

تصویر کامل حالت های ممکن و فضای حالت کامل به شکل گراف حالت در States.PNG

الگوریتم‌های جستجوی **ناآگاهانه**

(Uninformed Search)

1جستجوی اول-پهنا(BFS - Breadth-First Search)

ویژگی	مقدار
استراتژی	بررسی تمام گره‌های سطح n قبل از رفتن به سطحn+1
ساختار داده	صفFIFO - (Queue)
کامل بودن	بله - اگر راه حل وجود داشته باشد، حتماً پیدا می‌شود
بهینه بودن	بله - کوتاه‌ترین مسیر را پیدا می‌کند (با هزینه یکسان)
پیچیدگی زمانی	O(b^d)
پیچیدگی مکانی	O(b^d) تمام گره‌های یک سطح را نگه می‌دارد
نیاز به هیوریستیک	خیر

2جستجوی اول-عمق(DFS - Depth-First Search)

ویژگی	مقدار
استراتژی	رفتن به عمیق‌ترین گره قبل از بازگشت(Backtracking)
ساختار داده	پشتهLIFO - (Stack)
کامل بودن	خیر - ممکن است در حلقه‌های بی‌نهایت گیر کند
بهینه بودن	خیر - اولین راه حل پیدا شده لزوماً بهینه نیست
پیچیدگی زمانی	O(b^m)
پیچیدگی مکانی	O(bm) فقط مسیر فعلی را نگه می‌دارد
نیاز به هیوریستیک	خیر

3جستجوی تعمیق تدریجی(IDS - Iterative Deepening Search)

ویژگی	مقدار
استراتژی	ترکیب مزایای BFS و DFS - DFS با محدودیت عمق فزاینده
ساختار داده	پشته با محدودیت عمق
کامل بودن	بله - مانندBFS
بهینه بودن	بله - کوتاه‌ترین مسیر را پیدا می‌کند
پیچیدگی زمانی	O(b^d)
پیچیدگی مکانی	O(bd) بسیار کمتر ازBFS
نیاز به هیوریستیک	خیر

الگوریتم‌های جستجوی **آگاهانه**

(Informed Search)

4جستجوی(A-Star Search) A*

ویژگی	مقدار
استراتژی	بسط گره‌ای با کمترین f(n) = g(n) + h(n)
ساختار داده	صف اولویت‌دار(Priority Queue)
کامل بودن	بله - با هیوریستیک قابل قبول(admissible)
بهینه بودن	بله - با هیوریستیک سازگار(consistent)
پیچیدگی زمانی	وابسته به هیوریستیک - بهتر از جستجوی ناآگاهانه
پیچیدگی مکانی	O(b^d) تمام گره‌ها را نگه می‌دارد
نیاز به هیوریستیک	بله

5جستجوی اول-بهترین بازگشتی(RBFS - Recursive Best-First Search)

ویژگی	مقدار
استراتژی	شبیه A* اما با استفاده از بازگشت و محدودیت حافظه
ساختار داده	بازگشتی با نگهداری f-limit
کامل بودن	بله - با هیوریستیک قابل قبول
بهینه بودن	بله - با هیوریستیک قابل قبول
پیچیدگی زمانی	وابسته به هیوریستیک
پیچیدگی مکانی	O(bd) بسیار کمتر از A*
نیاز به هیوریستیک	بله