

تحلیل WUMPUS

تحلیل Wumpus World یک محیط مسئله است که برای مطالعه الگوریتم‌ها و روش‌های هوش مصنوعی طراحی شده است. Wumpus World معمولاً به عنوان یک محیط تصمیم‌گیری Markov استفاده می‌شود و به تست و ارزیابی الگوریتم‌ها و راه‌حل‌های هوش مصنوعی مورد استفاده قرار می‌گیرد. در Wumpus World، یک عامل (agent) در یک محیط قرار دارد که شامل یک شبکه مرتبط با اتاق‌ها، یک wumpus (هیولای خطرناک)، گنج و چاله‌ها است.

شرح محیط

اتاق‌ها - Wumpus - چاله - گنج

پارامترهای تحلیل

1. کامل بودن یک الگوریتم تحت Wumpus World کامل است اگر بتواند در هر شرایطی به صورت صحیح جواب دهد (یعنی گنج را پیدا کند و به خانه شروع بازگردد).
2. بهینگی الگوریتم بهینه است اگر کمترین تعداد حرکات را برای رسیدن به هدف انجام دهد.
3. پیچیدگی زمانی پیچیدگی زمانی الگوریتم بر حسب تعداد گام‌ها یا حرکات مورد نیاز برای حل مسئله است.
4. پیچیدگی فضایی پیچیدگی فضایی الگوریتم بر حسب مقدار حافظه مورد نیاز برای حل مسئله است.

نکات تحلیلی:

- الگوریتم‌های جستجو باید با هوشمندی حرکت کنند تا از مخاطرات مثل چاله‌ها و Wumpus جلوگیری کنند.

- تشخیص بوی Wumpus و وجود چاله‌ها نقش مهمی در بهینه‌سازی حرکات عامل ایفا می‌کند.

- الگوریتم‌ها باید واقعیت خود را با توجه به شرایط محیط به‌روز رسانی کنند.

- مواجهه با Wumpus باید با دقت انجام شود تا حمله نشوند و بهینه‌ترین مسیر به گنج را پیدا کنند.

```
class WumpusWorld:
```

```
    def __init__(self, size):
```

```
        self.size = size
```

```
        self.grid = [[None for _ in range(size)] for _ in range(size)]
```

```
    def move(self, direction):
```

```
    def shoot(self, direction):
```

```
    def grab(self):
```

```
    def perceive(self):
```

```
world = WumpusWorld(4)
```

```
world.move("right")
```

```
world.move("down")
```

```
world.perceive()
```