

❖ پیاده سازی مسئله 8 وزیر با استفاده از الگوریتم simulated annealing

الگوریتم SA در واقع بهبود یافته الگوریتم تپه نوردی (hill climbing) است با این تفاوت که به انتخاب های بدتر هم شانس ادامه می دهد تا شاید با کمی حرکت بد مارا به بهینه جهانی (global optimal) برساند.

الگوریتم به این صورت است که در مواجه با شرایط بد یعنی گره فرزندی که بدتر از گره والد باشد احتمال

$e^{\frac{\Delta E}{T}}$ را تولید می کند و فرزندی به صورت رندم تولید و در صورتی که در بازه احتمال باشند فرزند را گسترش می دهد. در ابتدا T عددی بزرگی فرض شده و با گذشت زمان به صفر رفته و دیگر گره فرزندی ادامه داده نمی شود.

$$\Delta E = \text{next_state} - \text{current_state} \quad \Rightarrow \quad 0 < e^{\frac{\Delta E}{T}} < 1$$

برای پیاده سازی از توابع زیر استفاده شده:

1. تابع printboard : برای ساختن صفحه بازی استفاده می شود که لیستی از جایگاه وزیر ها در صفحه را می دهد
2. تابع print_chess_board : صفحه بازی در هر مرحله را نمایش می دهد
3. تابع cost : برای محاسبه هزینه استفاده می شود که در این بازی هزینه بصورت تعداد وزیر هایی که یکدیگر را تهدید می کنند در نظر گرفته میشود.
4. تابع move : حرکات مجازی که توسط هریک از وزیر ها قابل اتخاذ است را نشان می دهد. این حرکات بصورت رندوم انتخاب می شوند و در صورتی اتخاذ می شوند که ویزر دیگری مورد تهدید قرار نگرفته و یا از صفحه بازی حذف نشود.
5. تابع simulated_annealing : برای پیاده سازی الگوریتم SA استفاده می شود.