

تمرین سری اول الگوریتم‌های مدرن در بهینه‌سازی

علی بنی‌اسد

۶ آبان ۱۴۰۱

۱ سوال اول

۱.۱ قسمت اول

۲.۱ قسمت دوم

۳.۱ قسمت سوم

۲ سوال دوم

مسئله QAP یکی از مسائل معروف در زمینه بهینه‌سازی ترکیبی است که به عنوان یکی از سخت‌ترین مسائل شناخته شده، در نظر گرفته می‌شود. این مسئله از آنجا که نقش بسیار مهمی در حل بسیاری از مسائل پیچیده روزمره ایفا میکند، در چندین دهه اخیر مورد توجه قرار گرفته است و بسیاری از محققین در زمینه‌های مختلف از جمله الکترونیک، آمار، اقتصاد و ... از آن بهره گرفته‌اند.

۱.۲ قسمت اول

یکی از مسأله‌های روزمره را می‌توان به صورت مسأله فروشنده دوره گرد مدل کرد. برای مثال، در یک روز n مکان مختلف در دانشگاه را بررسی کنم و کارهای مختلفی انجام دهم. هدف از این مسأله کم کردن زمان حرکت و با فرض حرکت در سرعت ثابت، می‌توان مسأله را به کمترین زمان نیز تبدیل کرد.

۲.۲ قسمت دوم

در این مسأله برای هر مسیر یک تابع هزینه محاسبه می‌شود و با توجه به شیب دانشگاه هزینه رفت بین دو مکان با برگشت آن مساوی نیست. از طرفی، برخی از مراکز را باید قبل از یک زمان مشخص بررسی کنم که یک قید به مسأله اضافه می‌کند. برای اضافه کردن قید به مدل، می‌توان زمانی که مرکز بسته است، تابع هزینه را به شدت بالا برد که مسأله هرگز از این قید عبور نکند. در نتیجه، n مرکز وجود دارد که تمامی باید بررسی شوند از طرفی این مراکز تا یک ساعت خاصی فعالیت دارند و با فرض قسمت قبل می‌توان زمان را به عنوان یک پارامتر بررسی کرد و متناسب با زمان برای هر مسیر کاست تابعی از زمان ارائه داد.

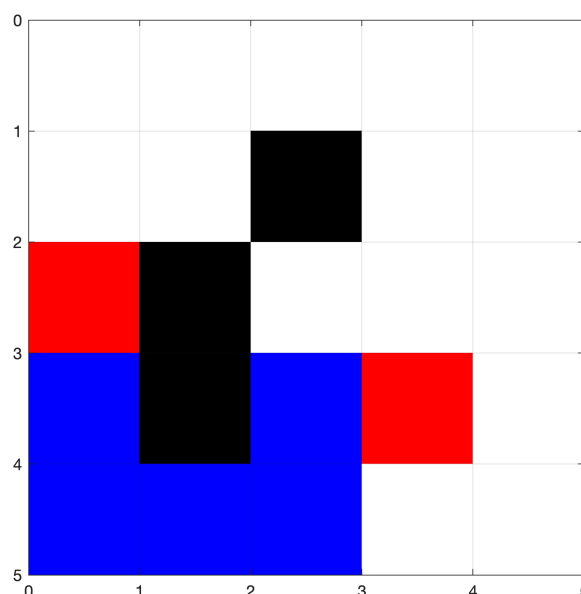
۳.۲ قسمت سوم

بله. با توجه به اینکه حل این مسأله از مرتبه $O(n!)$ است، حجم محاسبات با افزایش مراکز به شدت بالا می‌رود و دچار Combinatorial Explosion می‌شود.

۳ سوال سوم

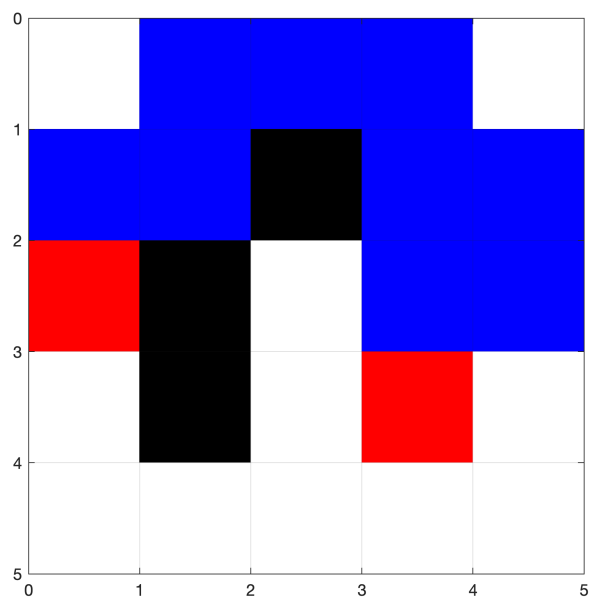
برای حل این مسأله ابتدا maze به یک گراف تبدیل می‌شود. در گراف هر خانه به خانه‌های اطراف متصل می‌شود به شرطی که مسیر مسدود نباشد. بعد از ساخت گراف برای پیدا کردن مسیر از الگوریتم Depth First Search (DFS) استفاده شده‌است. در این

الگوریتم به صورت بازگشتی از یک خانه به خانه دیگر می‌رود و ذخیره می‌کند که آیا از این خانه گذشته است یا خیر، که در مراحل دیگر سراغ خانه‌های تکراری نرود. در انتها بعد از رسیدن به مقصد مسیر طی شده را بر می‌گرداند. این روش تمامی مسیرهای از مبدأ به مقصد را بر می‌گرداند. با توجه به اینکه هزینه رفتن از هر خانه به خانه دیگر برابر یک (مسیر وزن‌دار نیست) است هزینه را می‌توان به صورت مستقیم خانه‌های طی شده را در نظر گرفت. کد پیاده‌سازی در الگوریتم DFS در فایل Q3.py آورده شده است. در مجموع ۷۶ مسیر برای رفتن از مبدأ به مقصد وجود دارد که تمامی مسیرها در فایل paths.csv آورده شده است. با توجه به اینکه بازه تابع هزینه از ۷ تا ۱۷ است نمی‌توان رتبه ۱ و ۵ را صرفاً یک مسیر مشخص معرفی کرد و حتی برای کمترین هزینه نیز دو مسیر وجود دارد؛ صرفاً مسیری که DFS بدست آورده و تابع هزینه آن رتبه‌های خواسته شده را داراست، آورده شده است و جواب یکتا ندارد ولی هزینه مسیر یکتا است.

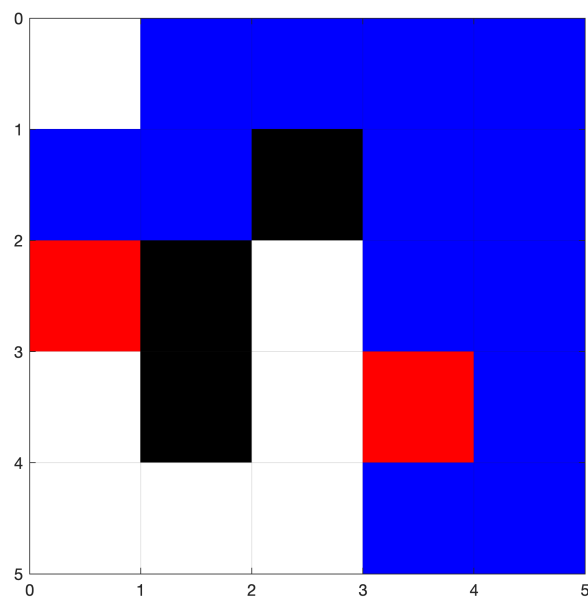


شکل ۱: مسیر با هزینه ۷ و رتبه ۱

انیمیشن مسیر حرکت نیز در فایل map_visualisation.m آورده شده است.



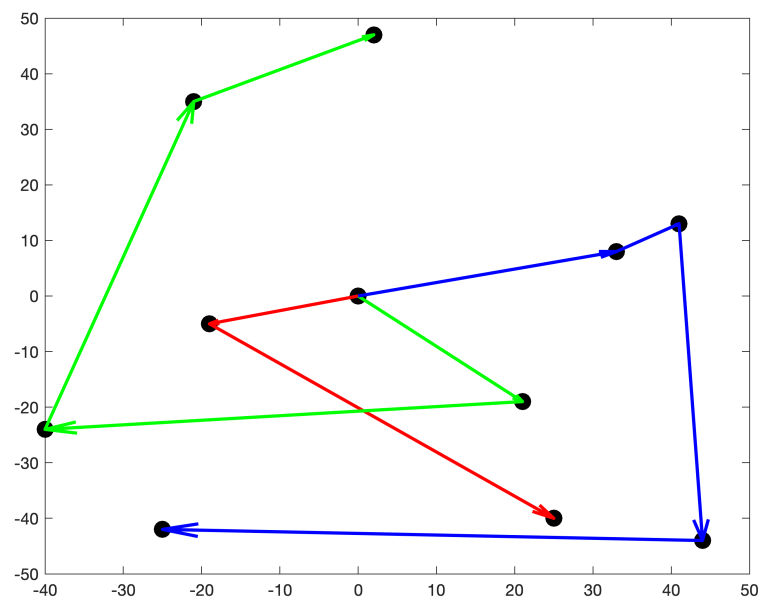
شکل ۲: مسیر با هزینه ۱۱ و رتبه ۱۰



شکل ۳: مسیر با هزینه ۱۵ و رتبه ۵۰

۴ سوال چهارم

برای حل این مسأله ابتدا به گروه بندی مراکز پرداخته شده‌است. در مرحله اول به این پرداخته شده است که هر ماشین باید به چند مرکز برود. برای بدست آوردن این گروه بندها حالت‌های تکراری پیش می‌آید به این صورت که گروه بندی (۶، ۲، ۲) و (۲، ۶، ۲) کاملاً یکسان است و تفاوتی نمی‌کند کامیونت اول به مرکز ۶ برود، دومی به مرکز ۲ و سومی به مرکز ۲ برود یا کامیونت اول به مرکز ۲ برود، دومی به مرکز ۶ و سومی به مرکز ۲ برود، صرفاً حجم محاسبات را بالا می‌برد. در نهایت ۱۴ حالت برای گروه بندی بدست آمد. در ادامه جایگشت‌های مختلف برای هر گروه بررسی شد. به این صورت، در یک آرایه بر اساس گروه بنده اعداد ۰، ۱ و ۲ گذاشته شد. جایگاه هر عدد نشان دهنده مرکز و مقدار هر جایگاه نشانده دهنده این بود با کدام کامیونت به آن مرکز رفت. بعد ساخت این آرایه تمامی جایگشت‌ها ساخته شد و هر جایگشت به صورت یک مسأله TSP بررسی شد. در انتها تابع هزینه به صورت بیشترین مسافت طی شده کامیونت‌ها در هر حالت قرار داده شد. کمترین زمان طی شده در شکل ۴ آورده شده‌است. تعداد کل حالات بررسی شده ۴،۰۸۵،۲۱۵ است. با فرض اینکه سرعت یک متر بر ثانیه است، کمترین زمان برابر با ۹،۲۶۵ ثانیه است.



شکل ۴: مسیر بهینه کامیونت‌ها به مراکز (هر رنگ نشان دهنده مسیر یک کامیونت است).