

Flow Shop ütemezés

Tabu keresés

Mesterséges intelligencia féléves feladat

2022/2023/1 félév

Nagy Róbert

Neptun kód: JMDRGG

2022.11.14

A feladat leírása:

Az erőforrások az ütemezési időszakban folyamatosan rendelkezésre állnak. Az erőforrások egyszerre csak egy munkán dolgoznak. A munkák legkorábbi indítási időpontja nulla (bármikor indíthatóak). Minden egyes munkához adott m számú operáció tartozik, melyeknek pontosan ismert a végrehajtási ideje: $p_{i,j}$, $i \in 1, 2, \dots, n$, $j \in 1, 2, \dots, m$. Az operációk végrehajtási sorrendje kötött és minden munka esetében azonos. Az operációk végrehajtása nem szakítható meg. A gépek között a munkák várakozhatnak, a műveletközi tárolók mérete nem korlátos. Az ütemezés célja az utolsóként elkészülő munka befejezési időpontjának minimalizálása.

Tabu keresés:

A keresés során bevezetünk egy tabu listát, melyben tároljuk, hogy mik azok a bázisok, ahol már jártunk. A tabulistához rendelünk egy maximális elemszámot, ha a tabulista megtelik, akkor kivesszük a legrégebbi elemet a listából.

A választott nyelv és könyvtár: Python 3.10, Numpy.

A megoldás lépései:

0. lépés: Definiálunk változókat, amelyek meghatározzák a maximális javulás nélküli iterációk számát, a tabu lista maximális elemszámát. Létrehozuk az üres tabu listát. Elindítunk egy számlálót, amely azokat az iterációkat számlálja, amelyekben a munkák befejezési ideje nem javult.

1.Lépés: Kezdetleges ütemezés generálása. A kezdetleges ütemezést lehet három módon lehet generálni. A legegyszerűbb módszer a munkák sorrendjének véletlenszerű keverése. A második módszer ütemezés generálása Novel hibrid algoritmussal. A leghosszabb, de legeredményesebb kiindulási ütemezést generáló módszer pedig a NEH (Nawaz, Ensore & Ham) algoritmus. Eltároljuk a kezdetleges sorrendet és annak befejezési idejét.

2.Lépés: Minden iterációban generálunk egy szomszédságot az eddigi legjobb sorozathoz.

3.Lépés: Megnézzük, hogy a generált szomszédságban lévő szomszédok benne vannak-e a tabu listában.

4.Lépés: Kiszámoljuk a szomszéd befejezési idejét, majd megnézzük, hogy az kisebb-e, mint az eddigi legjobb befejezési idő. Ha kisebb, akkor átállítjuk az eddigi legjobb változó értékét a szomszéd értékére. A szomszéd sorrendjét hozzáadjuk a tabu listához. Ha a tabu lista elérte a maximális méretet, akkor kivesszük a legrégebbi elemet a tabu listából.

5.Lépés: Megnézzük, hogy találtunk-e új legjobb értéket, ha igen, akkor visszaállítjuk a számlálót, egyébként inkrementáljuk azt.

6.Lépés: Következő iteráció. A ciklusnak akkor van vége, ha elérünk egy bizonyos iteráció számot, amelyekben nem volt javulás.

Felhasznált irodalmak:

https://en.wikipedia.org/wiki/Tabu_search

<https://arxiv.org/ftp/arxiv/papers/1407/1407.5931.pdf>

<https://reader.elsevier.com/reader/sd/pii/S0377221797001367?token=4E200F1C30848846AF6FC0E100AF131F452C39639A352506EE6D937A6EF7694B4BE1915DC7306102D93B1560FCAD6CCB&originRegion=eu-west-1&originCreation=20221114140721>

http://ait.iit.uni-miskolc.hu/~kulcsar/ET_2019_20_2f/ET_KGy_2019-20_2f_EA_11_Kereses_v04.pdf