Haladó Algoritmusok

Az eredeti tervben a függvény közelítést genetikus algoritmussal, az utazó ügynök problémát szimulált lehűtéssel, és a munka kiosztást hegymászó algoritmussal szerettem volna megoldani, de az egyes optimalizáló algoritmusokat könnyen meglehetett írni általános problémára, így végül mindhárom probléma bármely algoritmussal megoldható.

A hegymászó, és a szimulált lehűtés egy adott megoldás epszilon sugarú szomszédját használja fel a következő elem meghatározására, a genetikus algoritmus pedig a mutációt. Az epszilon sugarú szomszéd kiválasztása úgy történik, hogy az adott megoldás összes paraméterén egy kis mértékben változtatunk, még a mutáció esetében csak egyetlen paramétert változtatunk, de a változtatás a teljes tartományból kerül kiválasztásra. Sajnos a három problémából csak a függvény közelítés az ahol ez a két kiválasztási stratégia ténylegesen elkülönül, ugyanis az utazó ügynök, és a munka kiosztásnál csak egy módszert találtam, amely egy adott megoldást “lokálisan” tud megváltoztatni. Az utazó ügynöknél a gyakorlatban ez úgy történik, hogy két véletlenszerűen kiválasztott várost felcserélünk a bejárási sorrendben, a munka kiosztásnál pedig egy véletlenűl kiválasztott munkást ki-be “kapcsolunk”, azaz ha eddig használtuk az adott munkást akkor azt levesszük a listáról, és fordítva.

A függvény közelítésnél az epszilon sugarú szomszéd kiválasztása, úgy történik, hogy minden együtthatóhoz hozzáadunk egy kis számot [-1, 1] tartományból, még a mutáció csak egy együtthatót változtat meg, de azt bármelyik olyan számra beállíthatja amely benne van a konstruktorban megadott alsó-felső korlátban (ha nincsenek megadva korlátok a lehető legkisebb és legnagyobb float szám lesz a korlát).

Eredmények:

Az algoritmusok először egy kisméretű problémán lettek kiértékelve, ahol a globális optimumok manuálisan kiszámíthatóak. Optimalizáláskor a kilépési feltétel a maximum iteráció eléréséig, vagy a megadott fitness érték eléréséig tart. Először mindhárom algoritmust 1000 iterációig futtattam:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Szimulált Lehűtés | Genetikus Algoritmus | Hegymászó módszer |
| Függvény közelítés | **80.11** | 289.39 | 115.81 |
| Utazó ügynök | 33 | **29.4** | 41.8 |
| Munka kiosztás | **-123** | **-123** | **-123** |

A félkövér értékek az egyes problémát legjobban megoldó algoritmus fitnesz értéke, a zöld hátteres adatok a globális optimum értékei.

Ezek után a maximum iteráció számát 20 ezerre állítottam, amennyiben az algoritmus megtalálta a globális optimumot a futás előbb véget érhet. Az ehhez tartozó fitnesz értékek, és az egyes algoritmusok futási ideje miliszekundumban:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Szimulált Lehűtés | Genetikus Algoritmus | Hegymászó módszer |
| Függvény közelítés | 0.628 | 280.02 | **0.29** |
| Utazó ügynök | **19** | **19** | 43.8 |
| Munka kiosztás | **-123** | **-123** | **-123** |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Szimulált Lehűtés | Genetikus Algoritmus | Hegymászó módszer |
| Függvény közelítés | 376.6 | 10515.2 | **279.6** |
| Utazó ügynök | **50.8** | 77 | 81.6 |
| Munka kiosztás | 1 | 0.2 | **0** |

Itt a félkövér értékek a leggyorsabban lefutott módszer ideje.

A függvény közelítésnél a globális optimum nulla, azaz a szimulált lehűtés, és a hegymászó módszer is gyakorlatilag megtalálta ezt a pontot.

Az algoritmusok komplexebb problémákon is le lettek tesztelve, az utazó ügynök és a munka kiosztás 600 random várost/munkást tartalmaz, azonban a függvény közelítés csak annyiban lett nehezebb, hogy a mintavételezés kevésbé sürű, és a mintavételi pontok helye véletlenszerű. Az algoritmusok futási ideje nagyon megnő, ha az együtthatókat nem egy kisméretű intervallumból választjuk ki.

Chart

Description automatically generated

Az első néhány iterációt követő gyors fitnesz csökkenés után a genetikus algoritmus lokális optimumban ragad. A szimulált lehűtés kezdetben magasabb, majd alacsonyabb fitneszű értéket talál, de mivel logaritmikus a tengely felosztása, a két algoritmus közötti fitnesz különbség hiába tűnik nagynak, valójában mindkét érték egy nullához közeli szám.

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

Mind az utazó ügynök, mind a munka kiosztás során a genetikus algoritmus fitnesze gyorsabban csökken le. Az utazó ügynök esetében a genetikus algoritmus számottevően jobb megoldást talált, még a munka kiosztásnál mindhárom algoritmus ugyanoda konvergált, talán ez is a globális optimum.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Szimulált Lehűtés | Genetikus Algoritmus | Hegymászó módszer |
| Függvény közelítés | **0.42** | 308 | 0.89 |
| Utazó ügynök | 7782.27 | **4908.28** | 7926 |
| Munka kiosztás | **-51957** | **-51957** | **-51957** |