

Projektowanie efektywnych algorytmów - sprawozdanie z ćwiczenia 2

Marcel Guzik

6 stycznia 2022

Wykonawca: Marcel Guzik 256317
Prowadzący: dr inż. Jarosław Mierzwa
Grupa: Piątek 15:15 grupa późniejsza
Data realizacji zadania: 05.01.2022

1 Wprowadzenie

Celem ćwiczenia było zaimplementowanie algorytmu symulowanego wyżarzania dla asymetrycznego problemu komiwojażera.

2 Symulowane Wyżarzanie

Algorytm symulowanego wyżarzania to algorytm heurystyczny polegający na tym że na początku akceptujemy pewne rozwiązania które mogą być gorsze aby przejść większą przestrzeń rozwiązań przez co zmniejszamy szansę na utknięcie w minimum lokalnym.

Metoda symulowanego wyżarzania składa się z kilku części:

- Sposób wybierania sąsiada, czyli operacja move

Algorytm symulowanego wyżarzania działa poprzez stopniowe polepszanie aktualnego rozwiązania poprzez wybieranie rozwiązania sąsiedniego, tj. rozwiązania które w sposób minimalny różni się od rozwiązania poprzedniego. W problemie komiwojażera każde rozwiązanie to permutacja miast, a rozwiązania sąsiednie to rozwiązania w których dowolne dwa miasta są zamienione miejscami. Poruszanie się po przestrzeni rozwiązań w taki sposób, tj. ciągle przesuwając się do rozwiązań sąsiednich, stopniowo polepsza rozwiązanie, w nadziei na doprowadzenie w ten sposób do rozwiązania optymalnego.

- Temperatura oraz funkcja jej schładzania

Algorytm Symulowanego Wyżarzania różni się od zwykłego algorytmu zachłannego - w algorytmie występuje **temperatura**. Algorytm zaczyna pracę gdy temperatura znajduje się w swojej maksymalnej, startowej wartości - oznacza to że w wypadku gdy koszt nowego sąsiedniego rozwiązania jest wyższy niż koszt rozwiązania aktualnego, istnieje wysoka szansa że zostanie on zaakceptowany jako nowe rozwiązanie. Z kolejnymi iteracjami wartość temperatury zmniejsza się (co definiuje funkcja schładzania) i prawdopodobieństwo zaakceptowania gorszych rozwiązań sukcesywnie obniża się.

W programie stosowana jest następująca funkcja schładzania:

$$T_{i+1} = \alpha T_i$$

- Funkcja determinująca prawdopodobieństwo na przejście do sąsiedniego rozwiązania jeżeli to rozwiązanie jest gorsze od aktualnego

Prawdopodobieństwo przejścia z lepszego rozwiązania s do sąsiedniego, gorszego rozwiązania s_n określa funkcja prawdopodobieństwa $P(s, s_n, T)$, która wyznacza prawdopodobieństwo z różnicy kosztów obu stanów oraz aktualnej temperatury.

W programie występuje następująca funkcja prawdopodobieństwa:

$$P(s, s', T) = e^{-\left(\frac{s' - s}{T}\right)}$$

- Warunek stopu

Warunkiem stopu w programie jest spadek temperatury poniżej temperatury minimalnej t_{min} .

3 Kod programu

Program zrealizowano z wykorzystaniem programowania obiektowego. W tym celu utworzono trzy klasy: `Tsp`, `TspSolver`, oraz `SaTspSolver`.

3.1 Tsp

Klasa `Tsp` zawiera informacje reprezentujące instancję programu komiwojażera, tj. tablicę sąsiedztwa oraz ilość miast.

3.2 TspSolver

Klasa `TspSolver` jest klasą abstrakcyjną, definiującą interfejs pozwalający na uzyskanie rozwiązania dla danej instancji problemu komiwojażera. Dziedziczona jest ona przez inne klasy implementujące konkretne algorytmy rozwiązujące problem.

Jest to przykład wykorzystania wzorca projektowego strategii.

3.3 SaTspSolver

Klasa `SaTspSolver` dziedziczy po klasie `TspSolver` i implementuje funkcję `TspSolver::solve()` rozwiązującą problem komiwojażera używając algorytmu symulowanego wyżarzania (simulated annealing, SA).

4 Wyniki

Program kompilowano i testowano na systemie o następującej specyfikacji:

CPU: Ryzen 5 3600 6c/12t 3.6GHz 384KB/3MB/32MB cache

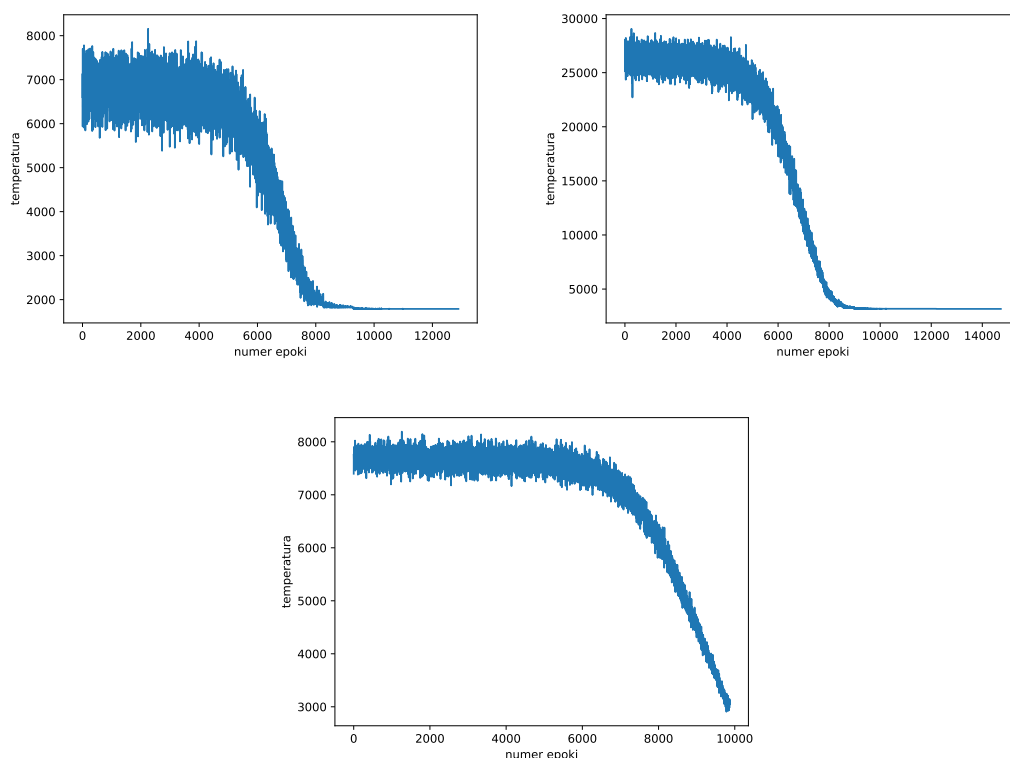
OS: Linux 5.14.21-2-MANJARO glibc: 2.23

skompilowano: `g++ -std=c++20 -g -O -Wall -Wextra -Wpedantic src/main.cpp -o zad1.out`

Dla każdej z przykładowych instancji program został uruchomiony 10 razy. Rozwiązanie najlepsze wybrano do porównania poniżej.

Nazwa instancji	limit czasowy	znaleziona ścieżka	najkrótsza ścieżka	błąd procentowy
ftv47.atsp	2m	1789	1776	0,7%
ftv170.atsp	4m	3166	2755	14,9%
rbg403.atsp	6m	2789	2465	13,1%

Tablica 1: Wyniki algorytmu dla trzech przykładowych grafów



Rysunek 1: Wykres kosztu aktualnego cyklu dla numeru iteracji działania algorytmu dla trzech powyższych grafów, w kolejności od lewej: ftv47.atsp, ftv170.atsp, rbg403.atsp

Znalezione ścieżki:

- ftv47.atsp

0 37 38 20 40 22 42 43 41 2 28 27 33 9 6 31 5 30 29 4 24 3 8 11 10 1 26 47 21 44 19 39 45 16
15 14 35 36 46 13 34 23 7 32 12 17 18 25 0

- ftv170.atsp

0 81 80 79 82 78 72 168 50 84 83 109 114 102 162 123 101 100 99 163 165 96 95 98 97 105
106 107 69 70 167 67 68 62 57 56 64 65 63 66 61 59 58 54 55 43 53 52 51 60 71 85 86 87 153
88 89 90 154 91 94 92 93 166 108 110 111 1 77 73 170 49 48 47 46 44 45 42 41 155 34 33 157
36 35 156 40 39 38 37 158 32 31 30 28 27 26 29 21 17 16 159 15 14 151 160 161 148 149 150
25 24 23 22 20 19 18 13 12 11 75 74 76 10 9 8 7 6 142 152 141 140 139 138 136 137 147 143

144 145 146 135 130 128 129 124 121 122 120 119 117 116 115 113 104 103 118 125 126 127
164 131 134 133 132 112 169 5 4 3 2 0

- rbg403.atasp

0 373 47 219 13 62 79 371 157 37 332 248 402 309 215 49 299 188 154 226 230 184 381 324
194 12 280 32 83 267 1 237 343 292 336 325 326 265 243 132 390 391 112 241 133 310 367
337 138 190 195 175 327 84 104 124 162 379 363 374 369 180 266 45 89 302 199 142 250 222
350 351 316 85 225 229 238 255 392 141 204 291 298 51 92 6 306 270 211 278 59 140 284 300
244 257 271 179 44 60 66 26 172 96 114 67 385 295 217 338 346 181 269 20 313 9 218 296
305 31 176 231 35 308 165 33 98 100 178 359 27 95 314 207 339 110 360 91 289 321 366 34
43 129 201 240 245 368 164 97 136 333 189 203 150 355 131 193 192 214 349 256 82 318 65
224 72 90 109 128 48 130 388 111 101 151 312 212 146 361 293 396 398 378 331 242 297 15 7
152 36 137 389 221 118 232 70 166 210 148 161 342 354 8 285 387 187 191 375 116 120 108
73 261 262 376 106 356 209 362 205 134 123 144 382 78 75 153 279 17 16 253 169 196 281
294 329 348 185 174 126 182 117 239 25 163 121 235 68 334 170 4 139 30 206 202 383 384 93
247 288 249 301 127 55 335 56 50 54 167 61 2 386 107 328 370 358 357 273 125 122 228 234
69 88 283 168 86 24 377 19 105 42 323 159 352 347 149 39 40 52 143 307 38 395 287 10 311
46 322 282 145 197 220 401 213 71 63 171 259 276 177 319 74 286 315 380 233 5 87 340 41
272 274 264 290 330 3 64 365 160 320 227 236 57 303 364 263 353 14 394 304 102 76 155 29
53 186 147 156 77 208 317 113 251 260 397 268 183 258 275 135 99 103 341 80 246 158 21
22 81 198 399 252 400 173 345 393 23 94 115 223 254 344 277 58 18 200 216 119 11 28 372

5 Podsumowanie

Symulowane wyżarzanie jest algorytmem dobrym dla większych instancji problemu komiwojażera. Możliwe że nie zwróci rozwiązania optymalnego, jednak dla dużych wielkości problemu (30 i więcej) złożoność obliczeniowa rośnie zbyt wysoko aby uzyskać optymalne rozwiązanie w rozsądnym czasie, zatem dla dużych instancji będziemy wspomagać się algorytmami opartymi o metaheurystyki, które mogą zwrócić rozwiązanie nieoptymalne, ale bardzo bliskie optymalnemu.