

Conclusões referentes à heurística do vizinho mais próximo com escolhas aleatórias (HVMPA)

Durante o processo de montar a tabela de comparações entre as heurísticas e juntamente com alguns pequenos testes feitos antes disso, foi observado que a HVMPA costuma funcionar melhor com α (alfa) que restringem os possíveis pontos, a serem escolhidos de forma aleatória, para uma quantidade muito pequena, evitando assim a escolha de soluções muito distantes da ótima ou até mesmo da heurística do vizinho mais próximo convencional. Dito isso, na primeira versão da tabela, por esse motivo e visando otimizar os resultados, ao invés de usar um α fixo, foi utilizado um valor que foi escolhido arbitrariamente de forma que restrinja o número de pontos a serem escolhidos. Abaixo segue a tabela com os valores que foram utilizados nesta primeira leitura e os intervalos em que foram utilizados, onde k corresponde a dimensão da instância analisada.

Intervalo	α
$k < 15$	0,15
$15 \leq k < 30$	0,1
$30 \leq k < 200$	0,05
$200 \leq k$	0,01

Com o α sendo escolhido desta forma a HVMPA apresentou resultados bem mais próximos à HVMP convencional, e em alguns casos, sendo a maioria destes em problemas pequenos, apresentou resultados melhores. Portanto, uma possível forma de melhorar os resultados da HVMPA seria encontrar um método para calcular estes alfas de forma que sempre restrinja o tamanho da escolha em apenas 2,3 ou 4 pontos, dando assim uma possibilidade de ser encontrado um resultado mais próximo que a solução ótima e ao mesmo tempo eliminando muitas péssimas escolhas que poderiam ser feitas caso haja mais possíveis escolhas. Outra forma seria atribuir um tamanho fixo para o número de possíveis escolhas, ou seja, invés de usarmos uma porcentagem do tamanho do problema usar algum valor dos que já foram citados anteriormente. Por fim, outra maneira de tentar otimizar esses resultados seria executar um determinado número de vezes a HVMPA, porém isto acarretaria em uma queda na eficiência do programa, principalmente em casos maiores.