



UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
OTIMIZAÇÃO COMBINATÓRIA E META-HEURÍSTICAS

2019/2

Trabalho Final

Marcos A. Spalenza

**Aplicação de *Particle Swarm Optimization* e
Clustering Search no Problema Capacitado de
Roteamento de Veículos**

Vitória, ES

2019

1 Introdução

O *Capacited Vehicle Routing Problem - CVRP*, é um problema clássico da literatura de otimização combinatória cujo propósito é atribuir valores ideais de rotas para os veículos de uma frota. Cada veículo é designado para executar a rota de entrega conforme sua capacidade de carga. Dada essa capacidade para cada rota e veículo, a alocação designa os melhores caminhos válidos de acordo com a capacidade limite e carga de entregas especificados. Assim, o processo de otimização deve atuar em duas linhas: a organização das cargas para os carros e a redução da rota de entrega.

Para o problema, a aplicação de meta-heurísticas visa interagir com as restrições de forma com que a combinação dos problemas indiquem gradativas melhorias nos resultados. Assim, em uma solução, a meta-heurística combina soluções de alocação dos n veículos que estejam de acordo com a capacidade c de carga partindo de um depósito d previamente estabelecido. Assim, a proposta é utilizar a meta-heurística *Particle Swarm Optimization - PSO* para o problema com intensificação das soluções através do *Clustering Search - CS**.

Neste trabalho, portanto, descrevemos a adaptação do *PSO* para problemas discretos, conhecido como *Discrete PSO - DPSO*. Paralelamente, dada a aproximação dos resultados do melhor item da população, mecanismo clássico deste algoritmo, a análise de vizinhança por *CS** foi aplicada como uma melhoria em potencial. Tal melhoria colhe os resultados da população do *DPSO* para aplicação de *Buscas Locais*.

Deste modo, este trabalho é descrito em 6 seções. Na Seção 2 indicamos a representação adotada para as soluções. Na Seção 3 descrevemos os detalhes da implementação e os movimentos da heurística. Na Seção 5 discutimos os testes e os resultados computacionais obtidos. Por fim, na Seção 6, apresentamos as conclusões da aplicação deste modelo no problema abordado.

2 Representação da Solução

Aplicado ao problema, o *PSO* foi adaptado como o modelo tradicional com vetores com os índices para cada um dos vértices do problema de alocação de rotas de veículos. Neste cada rota é indicada por partes de início e fim em um determinado ponto de depósito. Assim, considerando um vetor iniciado em um depósito 0, o início da próxima rota é indicada novamente por um ponto inicial em um depósito. Assim, uma possível solução de uma cadeia de 5 veículos atendendo 32 pontos de entrega é reproduzido por um vetor como o exemplo à seguir:

Solução S com 5 veículos V1, V2, V3, V4 e V5:

$S = 0, 7, 21, 31, 19, 17, 13, 0, 29, 18, 11, 4, 28, 23, 2, 3, 6, 0, 30, 26, 14, 24, 27, 0, 5, 20, 25, 10, 15, 22, 9, 8, 0, 16, 1, 12, 0$

$V1. = 0, 7, 21, 31, 19, 17, 13, 0$

$V2. = 0, 29, 18, 11, 4, 28, 23, 2, 3, 6, 0$

$V3. = 0, 30, 26, 14, 24, 27, 0$

$V4. = 0, 5, 20, 25, 10, 15, 22, 9, 8, 0$

$V5. = 0, 16, 1, 12, 0$

Como podemos observar, para uma solução S essa representação indica rotas pela repetição dos índices do vértice depósito. Cada veículo ($V1$, $V2$, $V3$, $V4$ e $V5$) sequencialmente passa pelos demais vértices a ele designados em ordem. A minimização fica por conta de estabelecer a menor rota para os veículos enquanto previamente é analisada sua viabilidade de acordo com o limite de carga. A resultante desses vetores no grafo de cidades pode ser visualizada através da representação gráfica apresentada pela Figura 1.

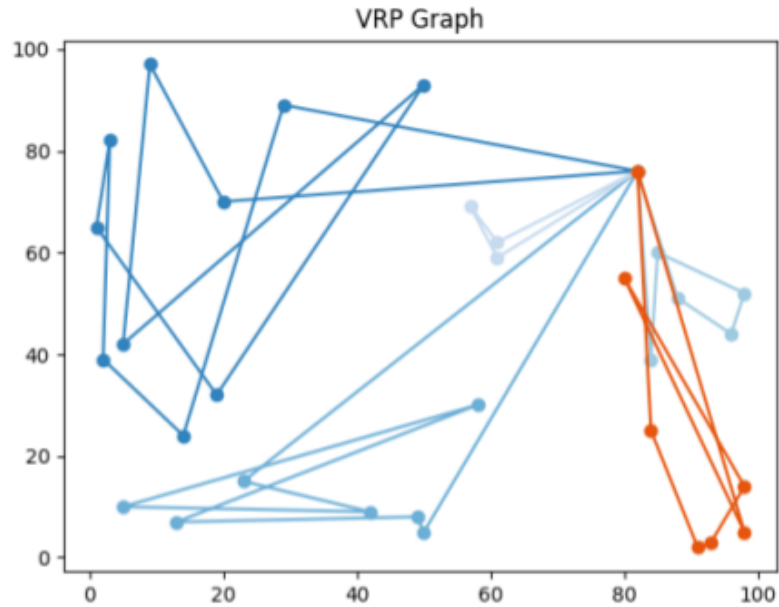


Figura 1 – Exemplo de solução realizada pelo sistema.

Como a Figura 1 apresenta, diferentes rotas são organizadas via meta-heurística visando a minimização do custo dentro da capacidade limite de carga. As mudanças do vetor de solução indicam reordenar as arestas utilizadas pelo sistema para atender a todos os vertices designados ao veículo. A ideia é estabelecer o balanceamento dos grupos atendidos por veículo ao mesmo tempo que ordena os vértices para encontrar o menor caminho possível.

3 Meta-Heurísticas

3.1 Discrete Particle Swarm Optimization - DPSO

O *Discrete Particle Swarm Optimization* (GONG et al., 2011), em geral segue o algoritmo clássico, adaptado para uso em problemas discretos. Inclui os ciclos de cálculo de velocidade, atualização da posição e controle populacional, mas sofre algumas adaptações. Inicialmente, a geração da população é dada pela separação aleatória dos vértices dentre o número de veículos disponíveis. Os vértices das rotas são embaralhados e adicionados iterativamente a um veículo, até que a carga deste esteja completa. Por fim as rotas são validadas para garantir que a solução é aplicável. Assim, assegura-se que os veículos estejam todos de acordo com as restrições na população inicial, divididos no número de veículos disponíveis.

Com a população os processos de modificação incluem atualizar a velocidade e a posição de cada partícula. Os passos de atualização incluem verificar a velocidade de forma similar ao modelo clássico, porém, a resultante é a probabilidade de mudança de cada posição. As posições do vetor que destoam muito do modelo do melhor valor encontrado em níveis global e local são atualizados de forma a serem mais suscetíveis a uma mudança. Esses valores de velocidade posicional são comparados com um bias aleatório que determina as posições que serão alteradas. Como o problema designado indica a troca de rotas específicas e não repetíveis, a atualização da posição parte por selecionar uma rota marcada para troca para alterá-la de posição com outra escolhida aleatoriamente. A resultante dessa troca passa pela validação de compatibilidade da carga, verificando se atende aos requisitos do problema antes de ser alterada de forma definitiva.

Assim, o processo de melhoria do *DPSO* é uma busca guiada, indicando pontos de possível melhoria na população. A ideia é que as regiões onde veículos são melhores alocados, como no melhor global influenciem as mudanças em rotas desorganizadas de forma a testar pontos distintos nessas posições específicas. O fator de velocidade perde um pouco de influência por inércia, ou seja, a solução deve analisar tendências a cada ciclo com menor impacto das seleções anteriores. Observa-se, portanto, que as posições com modificações em potencial sejam constantemente atualizadas.

3.2 Clustering Search

Em contrapartida com o *DPSO*, foi integrado o mecanismo de intensificação por clusterização, o *Clustering Search* (OLIVEIRA; CHAVES; LORENA, 2013). Bem como o *DPSO*, sua análise de vizinhança é estabelecida para a proximidade de soluções que indicam a possibilidade dos pontos médios representarem melhores buscas na vizinhança. Isso, porém, não é exatamente o modelo funcional para o problema adotado. Assim, tanto o bias de busca quanto o modelo de clusterização foi ajustado para dar significância ao

modelo.

A clusterização foi previamente definida para escolher arbitrariamente o número de clusters de acordo com o estado da população. Os dois testes foram feitos com o *Agglomerative Clustering* e com *Affinity Propagation*. O primeiro é um método hierárquico onde os clusters são formados pela similaridade dos itens. Enquanto isso, o segundo é um método de organização dos itens por comunicação, neste modelo a proximidade dos itens forma os cluster conforme a sensibilidade especificada para o algoritmo, chamada de *damping factor*.

Como a métrica euclideana é característica analisada no problema, e tradicionalmente aplicada a diversos problemas a mesma foi utilizada como medida de similaridade dos itens para a formação dos clusters. Após isso, as distâncias das soluções foram normalizadas para números entre 0 e 1 para verificar a equivalência dos itens nos grupos. A distância dos itens em geral manteve-se alta no final dos testes do *swarm*, com média de 0.8 de distância após a normalização, ou seja, com o número de iterações a população do *DPSO* ainda era divergente. Como é característico do método aglomerativo o crescimento incremental de clusters pela similaridade este primeiro foi descartado para uso prioritário do *Affinity Propagation*, priorizando não a equivalência dos clusters, mas sim a consistência dos itens em "vales" de busca.

Utilizando o método por afinidade, o *damping factor* foi calibrado por 0.7. Este valor varia de 0.5 a 1.0, sabendo que valores mais próximos de 0.5 são extremamente restritivos formando vários pequenos grupos e 1.0 representa o oposto. O fator médio forma um número intermediário de clusters e o fator de escolha de clusters torna-se necessário. Como a divergência entre os itens não é uma boa métrica para este caso, utilizamos o número de itens que foi agrupado. Grupos com menos de 5 itens, por possivelmente não se caracterizarem como próximos a pontos mínimos foram descartados.

Os testes incluíram o uso de Busca Local com 2-opt para os pontos máximos e mínimos do cluster. A partir do ponto médio do cluster, chamado centróide, virtualmente representado pela média dos itens agrupados, foram selecionados o item mais próximo e mais distante. Tais pontos representam os demais itens na tentativa de identificar se o ponto de mínimo é representado pelo centro ou pelo entorno do cluster formado. Adicional a essa busca, o melhor item da população também passa pela Busca Local.

Assim, o *Clustering Search* foi incrementado inicialmente no fim do processo de otimização e por fim, durante cada 50 ciclos do *DPSO*. Outra característica adicionada nessa combinação foi a especialização da busca pós clustering para adicionar algo deficitário no *swarm*. Durante as buscas locais do *DPSO*, as trocas irrestritas e com verificações de validade são aspectos de liberdade de busca das soluções. Porém, com o intuito de organizar as rotas em formas mais coerentes, a Busca Local do *Clustering Search* se faz complementar, modificando a posição de itens de uma mesma rota. Assim, a itensificação a

cada 50 ciclos visa organizar as rotas que foram selecionadas pela meta-heurística principal de forma a indicar arestas de melhor custo para os ciclos subsequentes.

4 Resultados

A meta-heurística descrita e o mecanismo de intensificação a ela acoplado foram executados cada qual 5 vezes. Cada instância foi executada paralelamente em um servidor com 32 GB de RAM e processador Intel(R) Xeon(R) CPU E5-2630v3 @ 2.40GHz com 32 cores. Os dados publicados por (AUGERAT et al., 1998; FISHER, 1994) foram utilizados para validação dos resultados. No total foram 53 *datasets* sintéticos divididos entre amostras A, B e F, de acordo com sua origem. Todos os datasets foram invariavelmente executados por 300 segundos.

Para o *DPSO*, testes prévios foram feitos para calibração e indicaram que a melhoria das soluções estava de acordo com as novas funções de velocidade e movimentação. Para isso foram testadas as populações de 50, 100, 500 e 1000 partículas e velocidades de 4, 2, 1, 0.1, 0.05. Foram selecionadas 500 partículas e fator de inércia de 0.1 para garantir mais de 2000 ciclos de teste por partícula e o filtro contínuo dos itens característico da modificação por velocidade *PSO*. Enquanto isso o *Affinity Propagation* foi calibrado para o *Clustering Search* produzindo clusters consistente. Os itens isolados em pequenos grupos (menores que 5) foram removidos dessa busca. Os resultados para cada instância, o número de iterações e a diferença para o resultado ótimo podem ser visualizados nas Tabelas 4 e 4.

Como podemos observar nas Tabelas 4 e 4, as soluções não foram as melhores de acordo com valores ótimos conhecidos, mas a adaptação apresentou resultados compatíveis. Com soluções e médias bem próximas, o *DPSO* foi inefetivo em três *datasets*, ao qual as melhorias em relação a população inicial foram de pequena escala **A-n61-k9**, **B-n57-k7** e **F-n135-k7**. Nas demais a convergência funcionou mais adequadamente e vários testes apresentaram o melhor item acima das 4000 iterações, próximo do limite de tempo especificado.

Com a adição do *Clustering Search* a intensificação dos resultados ocorreu e duas situações foram notadas. A primeira foi a melhoria dos resultados em relação ao *DPSO* em 30 dos 53 datasets. Além disso, o levantamento de iterações para melhoria indicado apresentou múltiplos de 50 em vários casos, indicando que a intensificação por força bruta nos itens relevantes da população representou o potencial da melhoria. A escolha do teste a cada 50 ciclos também representou apenas uma pequena redução em relação a meta-heurística original, importante para diversificação das soluções. Esses resultados ficam evidentes quando visualizados através das Figuras 2 e 3.

Como podemos observar nas Figuras 2 e 3 o mecanismo de intensificação realiza modificações na curva de melhoria. Essa característica é bem explícita com maiores

Instância	Ótimo	Melhor	Desvio	Iteração	Média	Desvio	Iteração
A-n32-k5	784.00	825.75	41.75	2100	839.89	55.89	2783
A-n33-k5	661.00	685.91	24.91	5907	694.60	33.60	2340
A-n33-k6	742.00	766.95	24.95	417	791.80	49.80	3128
A-n34-k5	778.00	790.76	12.76	3577	821.50	43.50	4453
A-n36-k5	799.00	822.72	23.72	4201	847.67	48.67	4826
A-n37-k5	669.00	745.44	76.44	6247	772.75	103.75	4400
A-n37-k6	949.00	1036.46	87.46	4134	1050.66	101.66	5705
A-n38-k5	730.00	839.85	109.85	4950	856.23	126.23	4788
A-n39-k5	822.00	934.05	112.05	4730	940.97	118.97	4111
A-n39-k6	831.00	899.64	68.64	5323	921.67	90.67	4755
A-n44-k6	937.00	1059.32	122.32	5167	1073.30	136.30	4411
A-n45-k6	944.00	1387.33	443.33	4189	1689.09	745.09	2220
A-n45-k7	1146.00	1244.93	98.93	6000	1261.14	115.14	4431
A-n46-k7	914.00	1037.41	123.41	4539	1051.81	137.81	4546
A-n48-k7	1073.00	1204.49	131.49	4624	1225.81	152.81	3925
A-n53-k7	1010.00	1267.14	257.14	3858	1297.61	287.61	5191
A-n54-k7	1167.00	1419.74	252.74	3414	1436.75	269.75	3974
A-n55-k9	1073.00	1313.48	240.48	4521	1350.47	277.47	4198
A-n60-k9	1408.00	1571.64	163.64	4826	1638.14	230.14	3934
A-n61-k9	1035.00	2558.46	1523.46	1789	2630.89	1595.89	3305
A-n62-k8	1290.00	1540.33	250.33	4395	1607.11	317.11	3184
A-n63-k9	1634.00	2161.26	527.26	4617	2235.78	601.78	3070
A-n63-k10	1315.00	1634.84	319.84	4290	1687.10	372.10	3107
A-n64-k9	1402.00	1704.42	302.42	4006	1774.02	372.02	3912
A-n65-k9	1177.00	1989.50	812.50	5044	2038.85	861.85	2324
A-n69-k9	1168.00	1583.21	415.21	1698	1656.09	488.09	3942
A-n80-k10	1764.00	2350.95	586.95	3905	2397.07	633.07	3765
B-n31-k5	672	679.67	7.67	3230	684.74	12.74	4041
B-n34-k5	788.00	799.66	11.66	3256	813.73	25.73	3030
B-n35-k5	955.00	987.31	32.31	4658	999.32	44.32	3759
B-n38-k6	805.00	822.21	17.21	3895	833.80	28.80	3186
B-n39-k5	549.00	580.96	31.96	3692	595.94	46.94	4761
B-n41-k6	829.00	891.05	62.05	2400	912.42	83.42	2971
B-n43-k6	742.00	789.89	47.89	5307	800.12	58.12	4200
B-n44-k7	909.00	1003.69	94.69	5099	1023.70	114.70	4044
B-n45-k5	751.00	917.74	166.74	2827	950.32	199.32	4247
B-n45-k6	678.00	917.85	239.85	1142	1019.07	341.07	2170
B-n50-k7	741.00	835.74	94.74	4213	865.46	124.46	4721
B-n50-k8	1313.00	1359.70	46.70	5928	1377.03	64.03	4346
B-n51-k7	1032.00	1525.41	493.41	5460	1576.16	544.16	4889
B-n52-k7	747.00	864.62	117.62	5231	885.11	138.11	3994
B-n56-k7	707.00	821.91	114.91	4977	850.04	143.04	3602
B-n57-k7	1153.00	2877.47	1724.47	1267	2974.13	1821.13	2649
B-n57-k9	1598.00	1742.25	144.25	5232	1778.52	180.52	4379
B-n63-k10	1537.00	1827.37	290.37	4102	1858.24	321.24	3674
B-n64-k9	861.00	1713.50	852.50	2167	2261.06	1400.06	2889
B-n66-k9	1374.00	1577.76	203.76	4904	1698.23	324.23	4190
B-n67-k10	1033.00	1329.75	296.75	2785	1354.12	321.12	2993
B-n68-k9	1304.00	1530.94	226.94	1491	1599.09	295.09	1985
B-n78-k10	1266.00	1698.36	432.36	2675	1744.69	478.69	3280
F-n45-k4	728	866.54	138.54	846	886.43	158.43	4255
F-n72-k4	238	423.08	185.08	4929	428.34	190.34	3504
F-n135-k7	1165	2.665.45	1.500.45	2420	2.791.49	1626.49	2360

Tabela 1 – Resultados obtidos após cinco execuções do *DPSO*.

Instância	Ótimo	Melhor	Desvio	Iteração	Média	Desvio	Iteração
A-n32-k5	784.00	834.38	50.38	5300	853.14	69.14	2.422
A-n33-k5	661.00	676.73	15.73	2200	693.52	32.52	3.847
A-n33-k6	742.00	759.96	17.96	4326	783.62	41.62	4.048
A-n34-k5	778.00	812.14	34.14	3412	827.04	49.04	4.039
A-n36-k5	799.00	856.39	57.39	4388	868.61	69.61	3.208
A-n37-k5	669.00	726.48	57.48	4480	748.96	79.96	3.869
A-n37-k6	949.00	1022.43	73.43	4100	1041.37	92.37	3.984
A-n38-k5	730.00	821.51	91.51	5250	851.46	121.46	4.475
A-n39-k5	822.00	933.37	111.37	5145	943.81	121.81	3.585
A-n39-k6	831.00	893.03	62.03	4400	916.02	85.02	3.874
A-n44-k6	937.00	1084.97	147.97	4729	1098.13	161.13	4.323
A-n45-k6	944.00	1417.27	473.27	3700	1639.76	695.76	3.730
A-n45-k7	1146.00	1187.19	41.19	3800	1239.04	93.04	3.622
A-n46-k7	914.00	1019.40	105.40	4677	1047.51	133.51	3.585
A-n48-k7	1073.00	1217.67	144.67	4400	1239.20	166.20	4.330
A-n53-k7	1010.00	1262.52	252.52	4450	1317.20	307.20	4.258
A-n54-k7	1167.00	1421.58	254.58	1050	1443.87	276.87	2.624
A-n55-k9	1073.00	1306.49	233.49	3750	1349.17	276.17	3.453
A-n60-k9	1408.00	1593.39	185.39	2550	1655.80	247.80	3.626
A-n61-k9	1035.00	2085.54	1050.54	450	2106.08	1071.08	1.520
A-n62-k8	1290.00	1568.98	278.98	3100	1610.52	320.52	3.406
A-n63-k9	1634.00	2149.95	515.95	4200	2279.23	645.23	2.475
A-n63-k10	1315.00	1692.09	377.09	2441	1712.59	397.59	2.828
A-n64-k9	1402.00	1806.58	404.58	3956	1820.53	418.53	3.544
A-n65-k9	1177.00	1982.67	805.67	2400	2110.52	933.52	2.690
A-n69-k9	1168.00	1651.08	483.08	3800	1688.04	520.04	3.419
A-n80-k10	1764.00	2374.36	610.36	3200	2490.59	726.59	3.008
B-n31-k5	672	678.57	6.57	4500	685.02	13.02	4.143
B-n34-k5	788.00	795.50	7.50	5050	808.04	20.04	3.815
B-n35-k5	955.00	982.24	27.24	4200	993.09	38.09	4.044
B-n38-k6	805.00	831.68	26.68	2180	833.60	28.60	3.463
B-n39-k5	549.00	571.60	22.60	5131	593.17	44.17	4.009
B-n41-k6	829.00	882.25	53.25	4533	913.25	84.25	4.467
B-n43-k6	742.00	792.63	50.63	5000	811.37	69.37	4.532
B-n44-k7	909.00	1005.62	96.62	2600	1017.31	108.31	3.887
B-n45-k5	751.00	850.12	99.12	1950	918.27	167.27	4.320
B-n45-k6	678.00	991.18	313.18	4400	1058.18	380.18	3.677
B-n50-k7	741.00	809.66	68.66	4500	880.89	139.89	4.106
B-n50-k8	1313.00	1365.61	52.61	4256	1395.65	82.65	4.055
B-n51-k7	1032.00	1385.96	353.96	4700	1587.69	555.69	4.388
B-n52-k7	747.00	856.57	109.57	3194	893.42	146.42	3.857
B-n56-k7	707.00	806.91	99.91	2300	852.92	145.92	3.530
B-n57-k7	1153.00	2288.17	1135.17	3950	2329.25	1176.25	3.240
B-n57-k9	1598.00	1774.02	176.02	3600	1802.08	204.08	3.887
B-n63-k10	1537.00	1823.82	286.82	3250	1862.75	325.75	3.672
B-n64-k9	861.00	1640.71	779.71	4000	1977.48	1116.48	2.628
B-n66-k9	1374.00	1681.26	307.26	3505	1743.62	369.62	3.534
B-n67-k10	1033.00	1324.86	291.86	3700	1368.87	335.87	3.595
B-n68-k9	1304.00	1590.26	286.26	3702	1641.47	337.47	3.149
B-n78-k10	1266.00	1740.62	474.62	3250	1807.11	541.11	3.036
F-n45-k4	728	830.52	102.52	3445	868.94	140.94	3.771
F-n72-k4	238	412.39	174.39	2418	424.04	186.04	2.567
F-n135-k7	1165	2.782.56	1.617.56	2750	2.903.01	1.738.01	2.196

Tabela 2 – Resultados obtidos após cinco execuções do *DPSO* com *CS*.

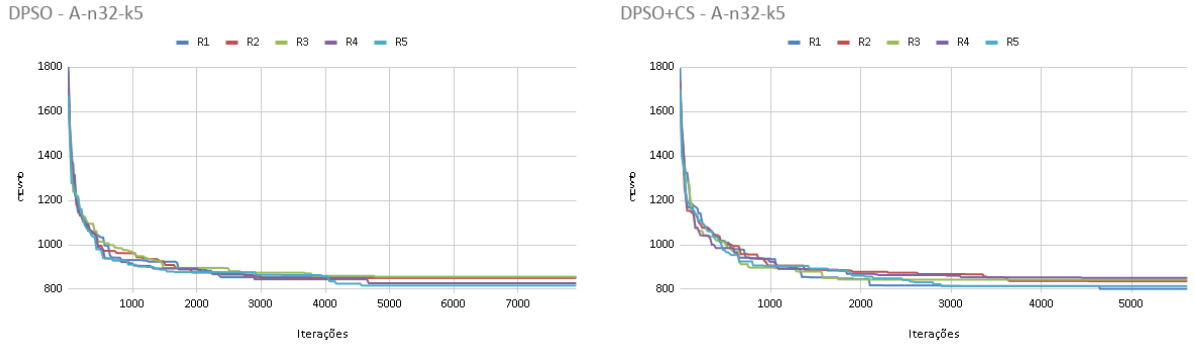


Figura 2 – Resultados obtidos com *DPSO* e *DPSO+CS* para a instância A-n32-k5.

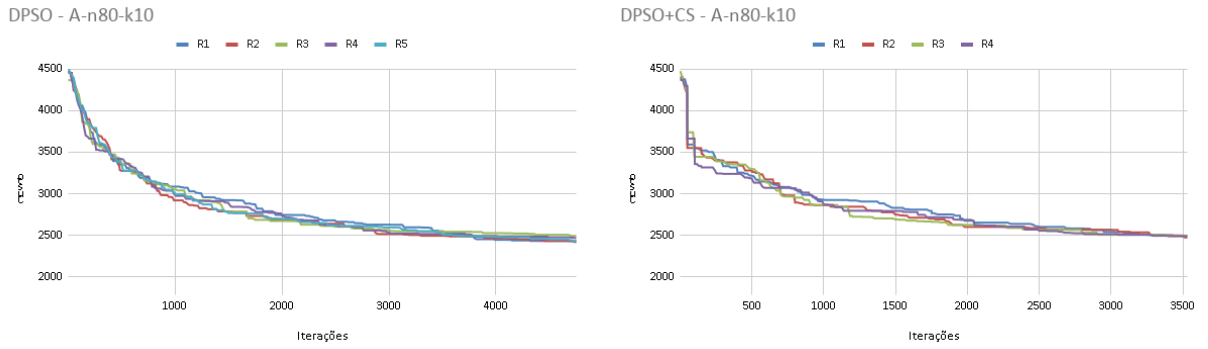


Figura 3 – Resultados obtidos com *DPSO* e *DPSO+CS* para a instância A-n80-k10.

distorções nas curvas a cada iteração, enquanto no *DPSO* o comportamento é aparentemente mais suave.

5 Conclusões

Os resultados obtidos foram minimamente adequados nas duas situações. Com leves problemas no *DPSO* clássico, o *Clustering Search* representou uma melhoria considerável em situações onde a movimentação pela vizinhança não apresentava melhora. Isso foi observado nos resultados das maiores instâncias. Outro fator relevante de se destacar é que nestas instâncias o comportamento do sistema aparenta ser interrompido durante o processo de melhoria inicial, o que apresenta certa lentidão para convergência aos mínimos.

Comparado com resultados da literatura, como o (AKHAND; PEYA; MURASE, 2017), o sistema demonstrou atingir seu objetivo. Foi competitivo e em determinadas instâncias foi equivalente a outros estudos de adaptação do *PSO*. O *Clustering Search* como adicional foi efetivo e apresentou melhoria na maioria das instâncias quando integrado ao *DPSO*. Por fim, como trabalhos futuros a perspectiva de integrar outros métodos ao movimento do *DPSO*, melhorando a influência da velocidade. É relevante ainda que um estudo minucioso da dispersão das soluções para que a clusterização represente melhor escolha de soluções para intensificação. Assim, como indicado na literatura, a combinação

com outros métodos pode ser fator relevante para chegar nos melhores resultados.

Referências

AKHAND, M.; PEYA, Z. J.; MURASE, K. Capacitated Vehicle Routing Problem Solving using Adaptive Sweep and Velocity Tentative PSO. *International Journal of Advanced Computer Science and Applications*, SCIENCE & INFORMATION, v. 8, n. 12, p. 288–295, 2017. Citado na página 8.

AUGERAT, P. et al. Separating capacity constraints in the cvrp using tabu search. *European Journal of Operational Research*, Elsevier, v. 106, n. 2-3, p. 546–557, 1998. Citado na página 5.

FISHER, M. L. Optimal Solution of Vehicle Routing Problems Using Minimum K-Trees. *Operations research*, INFORMS, v. 42, n. 4, p. 626–642, 1994. Citado na página 5.

GONG, Y.-J. et al. Optimizing the Vehicle Routing Problem with Time Windows: A Discrete Particle Swarm Optimization Approach. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, IEEE, v. 42, n. 2, p. 254–267, 2011. Citado na página 3.

OLIVEIRA, A. C. M. d.; CHAVES, A. A.; LORENA, L. A. N. Clustering search. *Pesquisa Operacional*, SciELO Brasil, v. 33, n. 1, p. 105–121, 2013. Citado na página 3.