

Isak Erikstad. 30.04.2025.

Metaheuristics – Final report

Call_7_Vehicle_3				
	Average objective	Best objective	Improvement (%)	Running Time
Final algorithm	1134176	1134176 [4, 4, 2, 2, 0, 7, 7, 0, 1, 5, 5, 3, 3, 1, 0, 6, 6]	65.02%	3.1

Call_18_Vehicle_5				
	Average objective	Best objective	Improvement (%)	Running Time
Final algorithm	2374420.0	2374420.0 [4, 4, 15, 15, 11, 11, 16, 16, 0, 6, 6, 5, 18, 5, 14, 17, 17, 14, 18, 0, 9, 8, 8, 9, 13, 13, 0, 7, 7, 3, 3, 10, 1, 10, 1, 0, 12, 12, 0, 2, 2]	73.50	259

Call_35_Vehicle_7				
	Average objective	Best objective	Improvement (%)	Running Time
Final algorithm	5163154.80	4964756 [34, 23, 23, 14, 34, 17, 14, 17, 33, 33, 0, 15, 15, 30, 7, 7, 30, 27, 27, 22, 22, 25, 25, 0, 19, 19, 24, 24, 13, 13, 28, 28, 31, 31, 0, 4, 4, 16, 16, 1, 3, 3, 21, 1, 21, 20, 20, 0, 8, 11, 11, 18, 18, 8, 5, 5, 2, 29, 2, 29, 0, 9, 6, 9, 35, 6, 35, 26, 32, 32, 26, 0, 12, 12, 10, 10, 0]	71.92%	2501

Call_80_Vehicle_20				
	Average objective	Best objective	Improvement (%)	Running Time
Final algorithm	11786169.30	11299737.00 [41, 70, 41, 11, 11, 45, 45, 69, 70, 48, 48, 69, 0, 18, 18, 20, 20, 0, 60, 60, 64, 64, 0, 4, 4, 68, 26, 68, 26, 0, 71, 71, 49, 49, 72, 73, 73, 10, 72, 10, 0, 54, 63, 54, 63, 5, 5, 77, 77, 0, 66, 66, 65, 12, 65, 12, 0, 39, 39, 55, 55, 80, 80, 0, 8, 8, 58, 58, 0, 15, 15, 23, 23, 61, 42, 67, 61, 42, 67, 28, 28, 0, 46, 3, 3, 36, 50, 46, 14, 36, 14, 50, 0, 34, 59, 34, 17, 59, 27, 17, 27, 33, 33, 0, 25, 25, 43, 79, 79, 43, 0, 9, 19, 9, 19, 47, 47, 6, 6, 0, 1, 21, 35, 1, 21, 35, 16, 7, 7, 16, 0, 32, 29, 32, 29, 0, 53, 30,	74.80%	7833

		30, 53, 76, 44, 76, 44, 52, 56, 56, 52, 0, 38, 51, 38, 78, 78, 51, 2, 2, 24, 24, 0, 40, 40, 13, 13, 37, 75, 37, 75, 0, 22, 62, 62, 74, 74, 22, 31, 31, 0, 57, 57]		
--	--	---	--	--

Call_130_Vehicle_40				
	Average objective	Best objective	Improvement (%)	Running Time
Final algorithm	18319040.30	17716174 [15, 15, 16, 16, 41, 41, 117, 117, 39, 39, 0, 74, 74, 34, 9, 34, 40, 9, 40, 0, 105, 105, 11, 11, 0, 60, 60, 102, 102, 90, 83, 83, 94, 90, 94, 0, 126, 126, 22, 22, 36, 53, 14, 53, 36, 14, 0, 96, 23, 96, 23, 62, 62, 18, 18, 0, 25, 25, 89, 111, 89, 111, 0, 73, 55, 73, 55, 110, 110, 64, 64, 0, 80, 80, 26, 26, 0, 21, 50, 50, 21, 0, 88, 88, 130, 130, 46, 10, 10, 30, 46, 30, 0, 123, 42, 123, 42, 118, 118, 28, 28, 0, 32, 82, 32, 82, 81, 91, 12, 91, 81, 12, 0, 49, 49, 33, 33, 125, 125, 48, 48, 0, 4, 4, 92, 68, 70, 92, 70, 68, 20, 20, 0, 103, 103, 59, 51, 59, 77, 51, 77, 0, 84, 98, 98, 114, 84, 65, 114, 127, 65, 127, 0, 3, 3, 121, 121, 120, 120, 104, 79, 79, 104, 0, 5, 5, 58, 95, 95, 58, 45, 17, 45, 17, 0, 97, 97, 0, 115, 113, 113, 115, 0, 19, 67, 67, 37, 19, 37, 128, 128, 0, 87, 69, 69, 87, 0, 106, 54, 106, 54, 0, 86, 86, 124, 99, 99, 124, 116, 116, 71, 71, 0, 66, 1, 66, 1, 0, 72, 27, 27, 112, 112, 72, 0, 2, 44, 2, 108, 108, 44, 0, 85, 85, 43, 43, 0, 52, 35, 52, 7, 7, 35, 0, 119, 119, 0, 129, 29, 29, 122, 129, 47, 47, 57, 122, 57, 0, 8, 8, 0, 100, 100, 107, 61, 109, 61, 107, 109, 0, 38, 38, 31, 63, 63, 31, 0, 78, 78, 0, 0, 75, 75, 0, 24, 24, 56, 56, 76, 76, 0, 93, 93, 13, 101, 13, 101, 0, 6, 6]	76.09%	12565

Call_300_Vehicle_90				
	Average objective	Best objective	Improvement (%)	Running Time
Final algorithm	44253953.60	42119254.0 [10, 10, 103, 103, 0, 29, 29, 246, 246, 27, 27, 0, 126, 126, 72, 131, 131, 72, 0, 210, 210, 0, 240, 42, 240, 42, 0, 273, 273, 41, 293, 41, 209, 293, 209, 0, 155, 155, 114, 257, 114, 257, 0, 0, 158, 158, 288, 56, 101, 56, 101, 288, 0, 15, 15, 271, 271, 0, 80, 80, 0, 54, 54, 0, 92, 92, 141, 148, 275, 275, 141, 148, 0, 74, 74, 0, 0, 185, 79, 14, 185, 79, 14, 0, 0, 11, 11, 0, 176, 63, 176, 63, 0, 287, 287, 104, 104, 0, 248, 248, 40, 40, 0, 195, 195, 5, 226, 226, 5, 22, 22, 0, 35, 194, 164, 35, 163, 194, 164, 292, 292, 163, 0, 0, 254, 224, 254, 224, 0, 25, 25, 181, 184, 181, 184, 0, 82, 208, 295, 208, 82, 124, 295, 124, 0, 206, 206, 0, 300, 77, 77, 125, 125, 300, 0, 0, 247, 247, 159, 159, 127, 127, 132, 132, 0, 123, 269, 269, 123, 290, 51, 290, 298, 51, 298, 0, 211, 266, 211, 266, 153, 153, 0, 276, 276, 47, 47, 84, 187, 187, 84, 0, 98, 98, 170, 170, 16, 16, 0, 215, 215, 149, 149, 198, 17, 17, 198, 0, 26, 26, 199, 81, 81, 199, 0, 218, 218, 145, 145, 0, 48, 225, 207, 207, 48, 225, 203, 112, 112, 203, 267, 267, 0, 150, 150, 32, 95, 32, 95, 251, 251, 0, 180, 180, 193, 193, 177, 177, 282, 282, 117, 117, 0, 133, 133, 274, 274, 119, 119, 33, 33, 0, 264, 83, 83, 281, 281, 182, 264, 3, 182, 3, 0, 30, 70, 30, 70, 166, 166, 50, 18, 50, 18, 0, 19, 19, 221, 122, 221, 122, 107, 107, 106, 106, 0, 286, 212, 291, 286, 212, 28, 291, 28, 231, 231, 0, 49, 49, 167, 167, 165, 146, 146, 165, 0, 278, 277, 277, 278, 7, 7, 75, 75, 0, 232, 39, 39, 232, 53, 34, 53, 34, 161, 161, 0, 67, 67, 204, 204, 191, 297, 191, 130, 297, 130, 0, 105, 270, 105, 270, 230, 230, 89, 115, 89, 115, 0, 147, 147, 62, 62, 64, 64, 0, 217, 217, 220, 189, 189, 220, 0, 261, 255, 261, 255, 173, 12, 173, 12, 0, 24, 24, 52, 52, 0, 168, 168, 142, 142,	74.09	26554

		262, 73, 262, 99, 99, 73, 0, 136, 136, 222, 222, 4, 4, 192, 192, 20, 20, 0, 234, 234, 21, 186, 21, 186, 0, 244, 244, 94, 94, 71, 71, 102, 140, 140, 102, 0, 263, 109, 200, 263, 109, 46, 200, 46, 8, 8, 0, 65, 229, 65, 283, 283, 229, 0, 31, 23, 23, 144, 31, 144, 205, 205, 0, 96, 188, 96, 188, 160, 160, 0, 169, 169, 252, 252, 0, 172, 69, 219, 227, 219, 172, 69, 227, 113, 258, 113, 258, 0, 201, 162, 162, 201, 259, 289, 289, 259, 0, 143, 143, 110, 110, 253, 120, 120, 55, 253, 55, 0, 116, 116, 268, 9, 60, 60, 9, 2, 2, 13, 268, 13, 0, 36, 88, 294, 36, 294, 88, 0, 280, 243, 280, 243, 100, 100, 0, 86, 86, 157, 197, 157, 197, 0, 152, 78, 78, 260, 152, 260, 0, 183, 235, 183, 235, 90, 296, 296, 90, 0, 76, 44, 44, 76, 85, 85, 93, 93, 0, 272, 61, 250, 61, 272, 250, 0, 233, 233, 202, 59, 59, 202, 190, 190, 0, 68, 265, 245, 68, 265, 58, 245, 139, 139, 58, 0, 279, 111, 279, 111, 0, 241, 237, 284, 237, 284, 241, 0, 249, 285, 249, 285, 0, 37, 121, 118, 118, 121, 175, 37, 175, 0, 137, 137, 228, 156, 228, 156, 0, 171, 178, 178, 171, 216, 216, 38, 38, 0, 66, 213, 66, 213, 0, 134, 134, 174, 174, 0, 1, 256, 256, 6, 6, 1, 0, 242, 242, 108, 108, 91, 128, 91, 128, 0, 179, 179, 238, 238, 154, 129, 129, 154, 0, 214, 43, 43, 214, 196, 97, 196, 97, 0, 135, 135, 151, 45, 236, 45, 151, 236, 0, 57, 57, 87, 87, 138, 138, 223, 223, 239, 239, 299, 299]	
--	--	--	--

The final experiment ran its full length (except for the last two which ran for 6000 iterations), but this took tens of hours to complete. For the final exam I will use multiprocessing computation and a time constraint inside the main loop to terminate search after 25-minute running time.

Algorithm explanation

Removal Operators:

1. Random removal - removes a random selection of calls
2. Shaw removal - removes calls that are like each other
3. Shaw removal with clipped range
4. Highest cost call removal - removes calls with highest costs
5. Highest cost call removal with clipped range
6. Random removal with clipped range
7. Exploration removal - used during exploration phases

Insertion Operators:

1. Greedy insertion - inserts calls at positions minimizing cost increase
2. Regret-2 insertion - considers opportunity cost of not placing calls
3. Regret-3 insertion - extended regret calculation with 3 positions
4. Regret-4 insertion - extended regret calculation with 4 positions

Other operators:

Vehicle swap

- Periodically attempts to improve solutions by swapping vehicles. Every 100th iteration.

- Applied to both incumbent and best solutions when improvements are found
- Will either do a 3-way swap or just switch 2

Algorithm Overview

This is an **adaptive weights algorithm** combined with **simulated annealing**. The key components are:

1. Adaptive Weighting Mechanism:

1. Tracks performance scores for each operator
2. Awards points based on solution quality (4 points for new best solution, 2 points for improvement, 1 point for new feasible solution)
3. Periodically updates operator weights based on performance scores
4. Normalized weights are used to select operators probabilistically

2. Simulated Annealing:

1. Uses temperature-based acceptance criterion for worsening moves
2. Temperature gradually decreases according to cooling schedule
3. Allows exploration of solution space early and focuses on exploitation later

3. Exploration Phases:

1. Periodically enters dedicated exploration phases using specialized removal operator
2. Switches between normal and exploration phases at defined intervals

4. Escape Mechanism:

1. Triggers when stuck for a certain number of iterations (escape threshold = 200)
2. Uses different strategies based on escape counter:
 - Increases weight of Random Removal Clipped operator
 - On third escape, removes ~50 calls and reinserts using greedy insertion
 - On second and fourth escape, reset incumbent to best solution found