Second report on the Tabu Search Algorithm

Fanny Kalinowski, Julien Molinier, Robin Lambert, Maxime Leras

24 Avril 2020

Sommaire

1 Introduction

Tabu Search is an algorithm created in 1986. It uses a deterministic heuristic on local search methods. The goal is to take a potential solution, that's to stay a local optimum, and to check its immediate neighbors in the hope of finding an improved solution. This time, some improvements of the Tabu Search algorithm have been made. This report presents the results and the conclusions of two improvements : the Random-based Tabu Duration and the Frequency-based Tabu Duration. At the end, the members of the team present their personal learning from this work. [1]

2 Reference results on 50 and 100 cities with constant Tabu duration

2.1 The two-best constant Tabu durations

In the first report, the two best constant Tabu durations we observed on the 50 cities problem were 20 and 80. They will be used in the following runs.

2.2 Run of the algorithm

The algorithm was run 10 times with two different configurations.
Configuration_1 of the first version :
Data set : 50 cities
Nb_iterations: 10 000
Duration_tabou: 20

[1] https://en.wikipedia.org/wiki/Tabu_search

Run	Best solution fitness	Iteration of the best solution	Nb_Local_minima
1	5693	340	3063
2	5649	323	2594
3	5649	1756	2592
4	5715	1691	2542
5	5658	325	2272
6	5649	533	2601
7	5766	781	2547
8	5756	122	1907
9	5644	603	2443
10	5851	52	2568

Table 1: Parameters found with the first version configuration at each run for Duration_Tabou equals to 20

Configuration_2 of the first version :
Data set : 50 cities
Nb_iterations: 10 000
Duration_tabou: 80

Run	Best solution fitness	Iteration of the best solution	Nb_Local_minima
1	5685	6890	2314
2	5655	261	2295
3	5644	214	2306
4	5645	8000	2269
5	5655	3449	2265
6	5649	2015	2200
7	5649	5226	2255
8	5649	9327	2270
9	5649	7228	2252
10	5649	3185	2293

Table 2: Parameters found with the first version configuration at each run for Duration_Tabou equals to 80

Configuration_3 of the second version :
Data set : 100 cities
Nb_iterations: 100 000
Duration_tabou: 20

Run	**Best solution fitness**	**Iteration of the best solution**	**Nb_Local_minima**
1	8115	328	44002
2	8151	424	44203
3	8063	380	44301
4	8346	227	44304
5	8193	376	44135
6	8351	132	44129
7	8295	1000	44099
8	8227	377	44622
9	8308	771	44002
10	8224	861	44170

Table 3: Parameters found with the second version configuration at each run for Duration_Tabou equals to 20

Configuration_4 of the second version :
Data set : 100 cities
Nb_iterations: 100 000
Duration_tabou: 80

Run	**Best solution fitness**	**Iteration of the best solution**	**Nb_Local_minima**
1	5269	744	21958
2	4944	5081	23233
3	5087	3168	24810
4	5017	2749	24634
5	4813	5406	24784
6	4823	19847	25840
7	5218	1609	23359
8	4911	6127	32819
9	4979	3874	29920
10	4855	13808	28593

Table 4: Parameters found with the second version configuration at each run for Duration_Tabou equals to 80

	Configuration_1	**Configuration_2**	**Configuration_3**	**Configuration_4**
Best solution fitness	5703	5652,9	8227,3	4991,16
Iteration of the best solution	652,6	4578,5	487,6	6241,3
Nb_Local_minima	5644	2271,9	44 196,7	25 995

Table 5: Average of the paramaters for the four configurations

For 10 runs in each configuration, we can see that the fitness is better with a Tabu duration of 80, for both 50 and 100 cities samples. However, the average iteration where the best fitness solution was found is clearly the lowest with a Tabu duration of 20, which means that the solution was found quicker than with 80, again for both cities samples.

What is quite interesting is that the best solution was found around the same iteration with a Tabu duration of 20 for both samples for each run (652,6 against 363,7). It is even a little bit faster for the 100 cities problem, but it's about the same rough size, in comparison of the 10 000 and 100 000 iterations respectively taken to run the algorithm. The maximum iteration was 1756, and strangely for the 50 cities problem. In both cases it appears that we never need respectively 10 000 and 100 000 iterations to find the best solution.

The number of local minimum visited are really important for the 100 000 iterations runs, of course. As expected, more local minimum were visited with a tabu duration of 80 for the 100 cities problem, than with the tabu duration of 20.

What we can say so far is that the 100 cities problem is a really complex one, and the best solution found was found only one time (4813 km). Let's see if the random-based tabu duration increase the results when it comes to fitness, but also performance-wise.

3 Random based tabu duration RTD

3.1 Running results

Run	Best solution fitness	Iteration of the best solution	Nb_Local_minima
1	5720	67	2664
2	5715	49	2561
3	5716	40	2384
4	5654	50	2580
5	5715	52	2629
6	5874	84	2653
7	5771	143	2402
8	5654	138	2496
9	5851	579	2432
10	5764	76	2429

Table 6: Results with RTD, alpha = 1, 10000 iterations and 50 cities

$$\begin{matrix} 10 & 4 & 4 & 6 & 6 & 0 & 8 & 0 & 6 & 8 \\ 7 & 5 & 5 & 7 & 4 & 4 & 9 & 7 & 4 & 4 \\ 4 & 4 & 10 & 0 & 10 & 0 & 5 & 4 & 4 & 4 \\ 0 & 4 & 4 & 4 & 4 & 4 & 4 & 7 & 6 & 0 \\ 5 & 8 & 5 & 4 & 4 & 4 & 4 & 4 & 0 & 4 \end{matrix} \qquad \text{(Mean Tabu duration by city table 6)}$$

Run	Best solution fitness	Iteration of the best solution	Nb_Local_minima
1	5644	3073	2249
2	5644	3489	2207
3	5644	9839	2184
4	5644	855	2240
5	5644	3241	2157
6	5644	651	2248
7	5644	7706	2184
8	5644	949	2232
9	5644	5027	2228
10	5644	4030	2238

Table 7: Results with RTD, alpha = 5, 10000 iterations and 50 cities

$$\begin{matrix} 38 & 18 & 18 & 18 & 18 & 18 & 18 & 18 & 18 & 18 \\ 18 & 18 & 18 & 18 & 18 & 18 & 18 & 18 & 18 & 18 \\ 18 & 18 & 18 & 18 & 18 & 18 & 18 & 18 & 18 & 18 \\ 18 & 18 & 18 & 18 & 18 & 18 & 18 & 18 & 24 & 34 \\ 23 & 27 & 38 & 18 & 18 & 18 & 18 & 18 & 37 & 18 \end{matrix} \qquad \text{(Mean Tabu duration by city table 7)}$$

Run	Best solution fitness	Iteration of the best solution	Nb_Local_minima
1	8349	351	24285
2	8319	137	24405
3	8413	97	23071
4	8652	56723	24592
5	8279	397	23167
6	8747	107	23713
7	8215	338	24323
8	8254	208	24384
9	8182	1176	23738
10	8287	710	24372

Table 8: Results with RTD, alpha = 1, 100000 iterations and 100 cities

$$\begin{matrix} 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 7 & 8 \\ 10 & 8 & 9 & 11 & 9 & 9 & 10 & 11 & 0 & 8 \\ 9 & 12 & 7 & 10 & 12 & 10 & 10 & 13 & 0 & 9 \\ 7 & 10 & 10 & 10 & 10 & 10 & 8 & 9 & 8 & 8 \\ 10 & 10 & 11 & 9 & 9 & 10 & 10 & 10 & 10 & 10 \\ 10 & 10 & 7 & 10 & 10 & 5 & 10 & 0 & 0 & 10 \\ 14 & 12 & 9 & 9 & 10 & 10 & 10 & 10 & 10 & 12 \\ 13 & 10 & 10 & 10 & 10 & 12 & 0 & 8 & 0 & 0 \\ 12 & 9 & 12 & 10.5 & 13 & 13 & 11 & 0 & 10 & 0 \\ 13 & 14 & 10 & 10 & 10 & 10 & 14 & 10 & 10 & 10 \end{matrix} \qquad \text{(Mean Tabu duration by city table 8)}$$

Run	Best solution fitness	Iteration of the best solution	Nb_Local_minima
1	7910	15848	22190
2	7949	74429	23207
3	8040	13322	20414
4	8247	1077	20931
5	8322	62656	19201
6	7974	42390	22730
7	8048	1900	20979
8	7944	2379	22143
9	8079	24546	21896
10	8175	18612	21571

Table 9: Results with RTD, alpha = 5, 100000 iterations_ and 100 cities

$$\begin{matrix} 50 & 50 & 50 & 50 & 50 & 50 & 50 & 50 & 50 & 49 \\ 50 & 50 & 50 & 50 & 50 & 50 & 50 & 51 & 51 & 50 \\ 49 & 50 & 50 & 50 & 50 & 50 & 50 & 51 & 51 & 50 \\ 49 & 50 & 50 & 50 & 51 & 55 & 50 & 50 & 50 & 50 \\ 49 & 50 & 50 & 50 & 51 & 50 & 50 & 50 & 48 & 46 \\ 51 & 51 & 46 & 50 & 50 & 51 & 49 & 50 & 50 & 50 \\ 49 & 50 & 50 & 51 & 51 & 51 & 50 & 50 & 51 & 50 \\ 50 & 50 & 50 & 50 & 49 & 49 & 46 & 48 & 50 & 50 \\ 50 & 50 & 52 & 51 & 50 & 49 & 49 & 50 & 50 & 50 \\ 50 & 49 & 50 & 50 & 50 & 50 & 50 & 50 & 50 & 50 \end{matrix} \qquad \text{(Mean Tabu duration by city table 9)}$$

	Configuration_6	**Configuration_7**	**Configuration_8**	**Configuration_9**
Best solution fitness	5743,4	5644	8369,7	8060,8
Iteration of the best solution	127.8	3886	6024,6	25715,9
Nb_Local_minima	2523	2216,7	24004,8	21526,2

Table 10: Average of the paramaters for the four configurations

We can observe that the solution is better for the configurations 7 and 9, that's to say when alpha = 5 for the 50 and the 100 set of cities.

Let's calculate the bounds. When alpha = 1, lower_bound = 158 and upper_bound = 474. Consequently, the tabu duration is included in S1 = [158:474]. There are 316 values in this set, so the tabu duration can take 316 random values.

When alpha = 5, lower_bound = 790 and upper_bound = 2 371. The tabu duration is included in S2 = [790:2 371]. There are 1581 values in this set, so the tabu duration can take 1581 random values.

Regarding S1 and S2, the probability to pick a high random value for the tabu_duration is higher in S2 than in S1. The more high tabu_duration is, the more visited are the neigbors of the solution because it prohibits to reeuse the same neighbors in the solution.

So using alpha = 5 raise the probability to find a neighbor more intersting than the best solution. More neighbors are explored, 26 000 for the configuration_9 against 23 000 for the configuration_8.

The randomness isn't predictable, we can't really make assured conclusions. Nevertheless, we can observe that using the randomness, the best solution is improved comparing to the configuration in the previous part. This is due to the fact that 20 and 80 are not even included in S1 and S2. Every random value picked from those two sets will produce a higher tabu duration that 20 or 80. So, more neighbors are explored. This is why the best solution is improved.

4 Frequency-based tabu duration FTD

4.1 Run the algorithm for the realease FTD1

Run	**Best solution fitness**	**Iteration of the best solution**	**Nb_Local_minima**
1	5649	8100	2525
2	5773	154	2425
3	5645	2015	2478
4	5771	358	2495
5	5768	2716	2364
6	5839	1023	2513
7	5715	1434	2421
8	5839	698	2493
9	5654	8093	2562
10	5768	2555	2236

Table 11: Results with FTD1, alpha = 1, 10 000 iterations_ and 50 cities

$$\begin{matrix} 10 & 16.45 & 10 & 10 & 0 & 10.30 & 14.99 & 15.52 & 10.67 & 10.66 \\ 10.57 & 10.77 & 10.57 & 10 & 10.18 & 14.07 & 12.54 & 14.04 & 11.70 & 10.54 \\ 10.73 & 10.73 & 13.41 & 13.50 & 11.01 & 11.49 & 11.59 & 10 & 11.79 & 11.54 \\ 11.57 & 11.63 & 11.90 & 11.54 & 10 & 12.23 & 11.81 & 10.62 & 11.58 & 11.46 \\ 10.85 & 10.27 & 10 & 10 & 10.34 & 10.34 & 15.87 & 15.86 & 10 & 16.47 \end{matrix}$$

(Mean Tabu Duration for each city (alpha = 1 and 50 cities))

Above, the matrix shows the mean tabu duration for each city during one of the runs with FTD1 where there are 50 cities and alpha is 1. We can see that some cities weren't visited enough times to have a Tabu duration changed so the mean value is 10 (same as default constant). We can also see that on 10 000 iterations, one city hasn't been visited once so the mean value is 0.

Run	Best solution fitness	Iteration of the best solution	Nb_Local_minima
1	5768	8725	2273
2	5750	7432	2369
3	5644	9855	2402
4	5644	140	2320
5	5644	174	2349
6	5644	6833	2351
7	5786	8073	2293
8	5723	3714	2389
9	5763	7582	2177
10	5644	9468	2370

Table 12: Results with FTD1, alpha = 5, 10 000 iterations_ and 50 cities

10	29.06	15.66	11.66	12.75	13.61	18.43	18.69	15.71	15.70
15.52	15.98	14.86	11.27	15.72	19.20	18.55	18.51	17.44	17.84
18.71	10.04	15.30	20.54	20.62	18.35	20.92	20.42	10.23	10.14
10	12.21	15.67	22.67	24.03	23.14	19.26	12.41	10	10
0	10	10	11.21	16.36	16.10	31.50	31.55	10	33.5

(Mean Tabu Duration for each city (alpha = 5 and 50 cities))

Run	Best solution fitness	Iteration of the best solution	Nb_Local_minima
1	8502	64479	23907
2	8468	87330	17992
3	8089	51183	22372
4	8096	80233	23466
5	8356	83545	23729
6	8311	6760	22647
7	8310	94709	22311
8	8359	74478	22969
9	8287	59608	23871
10	7994	59341	24091

Table 13: Results with FTD1, alpha = 1, 100 000 iterations_ and 100 cities

11.02	33.30	10	10	10	10	10	0	22.94	23.01
19.11	23.36	25.40	25.96	26.39	14.59	10	10	0	10
11.20	17.70	17.26	16.94	17.93	10	10	10	10	12.79
12.79	15.71	16.50	16.47	15.48	14.19	13.33	12.94	11.62	10
10	23.95	24.51	18.81	17.65	15.05	11.10	11.27	15.88	16.94
14.67	17.21	18.61	10	13.18	13.17	10	10	17.22	17.23
18.60	15.60	14.94	15.03	16.65	16.06	10	10	10	10
10	10	10	10	10	10	13.69	13.70	13.69	12.03
19.09	19.09	14.11	10	10	10	10	10	10	10
10	10	15.32	15.33	10	10	11.11	31.4187	32.95	31.97

(Mean Tabu Duration for each city (alpha = 1 and 100 cities))

Run	Best solution fitness	Iteration of the best solution	Nb_Local_minima
1	8703	91598	22382
2	8045	74418	19044
3	8217	11256	20928
4	8329	92618	23365
5	8323	92495	21745
6	8068	61046	21696
7	7910	77713	22646
8	8039	56475	22773
9	8061	47198	22537
10	8035	8688	22771

Table 14: Results with FTD1, alpha = 5, 100 000 iterations_ and 100 cities

40.61	62.88	14.13	23.90	32.93	63.74	64.65	59.97	12.59	12.48
13.21	46.64	45.78	39.87	45.44	42.48	42.10	11.02	11.20	14.44
14.11	43.89	44.65	10.30	10	57.90	54.67	13.84	16.55	20.97
44.02	44.59	16.94	14.48	26.44	28.90	29.67	24.05	16.43	18.80
32.31	43.34	40.56	24.18	30.60	12.23	44.20	46.63	54.30	54.30
19.22	18.55	17.45	18.76	23.86	24.33	24.33	22.19	24.57	27.70
22.68	16.77	19.07	19.76	19.18	22.48	28.49	10	29.91	30.59
25.91	23.05	23.19	26.75	22.78	15.16	10	10	17.63	16.55
24.18	29.32	10	28.43	24.07	11.30	11.80	17.27	25.84	24.59
27.48	18.37	48.30	44.63	20.06	13.08	25.61	42.44	41.27	58.40

(Mean Tabu Duration for each city (alpha = 5 and 100 cities))

	Configuration_11	**Configuration_12**	**Configuration_13**	**Configuration_14**
Best solution fitness	5 742,1	5 701	8 277,2	8 173
Iteration of the best solution	2 714.6	6 199.6	66 166,6	61 350,5
Nb_Local_minima	2 451.2	2 329,3	22 735,5	21 988,7

Table 15: Average of the results for the four runs with FTD1

4.2 Algorithm performances for FTD1

For both cities sizes, we can see that having a greater alpha gives better results. This is due to the fact that the effect of changing the tabu duration depending on frequency is much more visible when alpha is big.

When we see the best solution fitness average, we see a little increase in performances when alpha is greater. But when we look at the fitness one by one, optimal solutions are much more present when alpha is 5 than when alpha is equal to 1. For example, the optimal solution (5644km) for 50 cities is found 0 times with alpha = 1 but 5 times with alpha = 5. The same thing can be noticed for the 100 cities problem where a solution of 7910km was found with alpha = 5.

In the mean tabu duration for each city matrix, we can see that when alpha increase, the tabu duration for cities that where often visited is much higher. This is the only difference thing that changes when alpha changes and it explains the different results.

4.3 Run the algorithm for the release FTD2

tab pour les 10 000

Run	**Best solution fitness**	**Iteration of the best solution**	**Nb_Local_minima**
1	5645	179	2414
2	5763	9779	2256
3	5811	9853	2378
4	5763	55	2250
5	5649	9728	2560
6	5645	45	2416
7	5649	8467	2294
8	5715	6360	2356
9	5644	4067	2494
10	5795	7583	2339

Table 16: Results with FTD2, alpha = 1, 10 000 iterations_ and 50 cities

15.5924	14.9471	14.89	14.0797	13.7367	16.485	16.6174	16.3766	16.8132	14.841
13.7467	11.4583	12.4524	16.2902	16.9431	16.3834	16.1463	16.7192	11.8333	13.6667
12.3636	12.7309	18.9462	21.9	21.7714	21.599	24.504	23.7905	17.8571	11
11	11	11	11.9481	14.1354	13.9975	15.3799	15.3709	11	19.541

(Mean Tabu Duration for each city (alpha = 1 and 50 cities))

Run	**Best solution fitness**	**Iteration of the best solution**	**Nb_Local_minima**
1	5644	4261	2267
2	5644	7975	2219
3	5645	663	2231
4	5655	8013	2277
5	5645	367	2249
6	5644	106	2267
7	5644	2390	2290
8	5644	7043	2251
9	5644	3771	2316
10	5644	2702	2320

Table 17: Results with FTD2, alpha = 5, 10 000 iterations_ and 50 cities

11.2308	41.9507	32.2477	35.6451	34.562	34.5741	36.3537	40.0609	39.6876	39.1119
38.1073	40.4421	38.0241	37.9798	35.3386	36.5734	36.6914	37.3024	35.4751	34.0413
24.4427	21.5294	30.5531	36.0771	37.8721	38.5858	36.3602	34.6167	34.1869	30.1285
25.4908	27.0628	39.9442	44.9256	45.6255	45.6936	43.5129	39.8516	45.2051	29.3
28.625	27.9189	24.4375	23.912	27.3402	27.3573	24.0848	24.3129	11	35.4404

(Mean Tabu Duration for each city (alpha = 5 and 50 cities))

Run	**Best solution fitness**	**Iteration of the best solution**	**Nb_Local_minima**
1	8319	38314	23490
2	8279	45937	21257
3	7944	92926	20264
4	8585	68202	22017
5	8323	27327	23669
6	8246	98642	22213
7	8248	51592	20030
8	8292	78529	22483
9	8410	29825	23301
10	8358	63426	18636

Table 18: Results with FTD2, alpha = 1, 100 000 iterations_ and 100 cities

14.2213	33.1088	44.2921	11	11.1667	15.6803	16.3703	19.4014	26.2325	25.8807
11.85	11.8417	12.5427	43.2813	43.6433	45.8751	13.2413	12.0173	12.3448	12.7016
13.2642	0	11.7448	11.7448	11.8506	11.7725	11.8314	11.8209	11	11
11	11	11	13.4271	12.5854	13.5958	11.1515	28.1966	32.108	36.2973
36.3823	37.6224	28.1438	26.9951	31.9657	30.9723	34.3475	30.9577	22.5665	27.2117
27.1805	27.3557	25.5466	25.2069	19.3455	19.7077	29.5313	31.7773	27.3007	29.4811
18.3236	17.57	14.7732	14.184	11.7553	30.7526	30.4278	31.9126	11	13.3327
13.184	12.875	11.1818	24.764	33.3649	30.6882	31.5677	25.0468	25.6973	11
26.3582	38.1855	37.2955	39.0673	36.7252	13.4788	13.2473	13.2207	0	11
11	12.0518	12.0552	18.1671	19.1905	23.4462	17.3333	13.6648	14.103	26.9156

(Mean Tabu Duration for each city (alpha = 1 and 100 cities))

Run	**Best solution fitness**	**Iteration of the best solution**	**Nb_Local_minima**
1	8054	53160	21517
2	7949	98104	20949
3	7911	84982	20983
4	7956	99565	21208
5	8256	58593	19764
6	7955	94970	19999
7	7959	95487	18694
8	8183	13473	19997
9	7948	79910	19707
10	8189	88352	19564

Table 19: Results with FTD2, alpha = 5, 100 000 iterations_ and 100 cities

68.1509	112.007	101.918	98.6314	105.809	96.8933	100.215	119.445	102.729	94.9213
83.724	92.953	99.9827	102.834	107.271	107.257	107.438	86.8929	79.7498	72.0014
66.5573	66.7006	64.6535	68.838	48.5701	38.8745	40.5317	31.665	44.4055	103.265
94.904	89.928	96.1262	71.4068	87.5286	89.8988	89.9572	95.5804	99.7745	62.5995
32.7234	60.2167	61.1399	61.2695	63.1004	40.3795	32.8901	30.3592	34.6225	46.0168
49.5236	51.6692	50.1898	34.7083	26.3168	21.9012	24.4254	27.8591	31.8502	31.199
28.2871	30.2453	33.5564	37.9226	38.6231	39.0285	40.2019	41.1092	44.898	41.8697
43.2374	43.1569	34.0033	27.1339	23.6916	23.4921	31.9303	37.0714	43.0826	58.7028
85.291	86.2254	84.4447	86.1797	75.2931	74.6028	60.3596	54.8754	54.2978	49.8056
52.1828	48.3708	45.3021	60.7538	62.2397	63.7988	67.8385	72.2626	89.3126	82.0704

(Mean Tabu Duration for each city (alpha = 5 and 100 cities))