



In [4]:

```

import qhea
import benchmark
problems = benchmark.get_names()
population_size = 24
generation_count = 48
executions = 1000
params = [
    qhea.Parameters(
        population_size=population_size,
        generation_count=generation_count,
        local_strategy=qhea.LocalOptimization.SimAnnealingThermalRandom,
        selection_strategy=qhea.Selection.Ranking,
        selection_pressure=1.5,
    ),
]
names = ['Ranking medium pressure with local']
benchmark.analyze_and_display(params, executions, problems, names)

```

```

=====
=====

```

Algorithm was tested with 1 different parameter configurations
 For each parameter configuration, the algorithm was tested against these problems:

```
['random-30', 'tsp-5-25-10-20', 'tsp-5-25-10-100', 'tsp-5-25-1-5']
```

For each problem, the algorithm was executed 1000 times to collect data.

```

-----
-----
-----

```

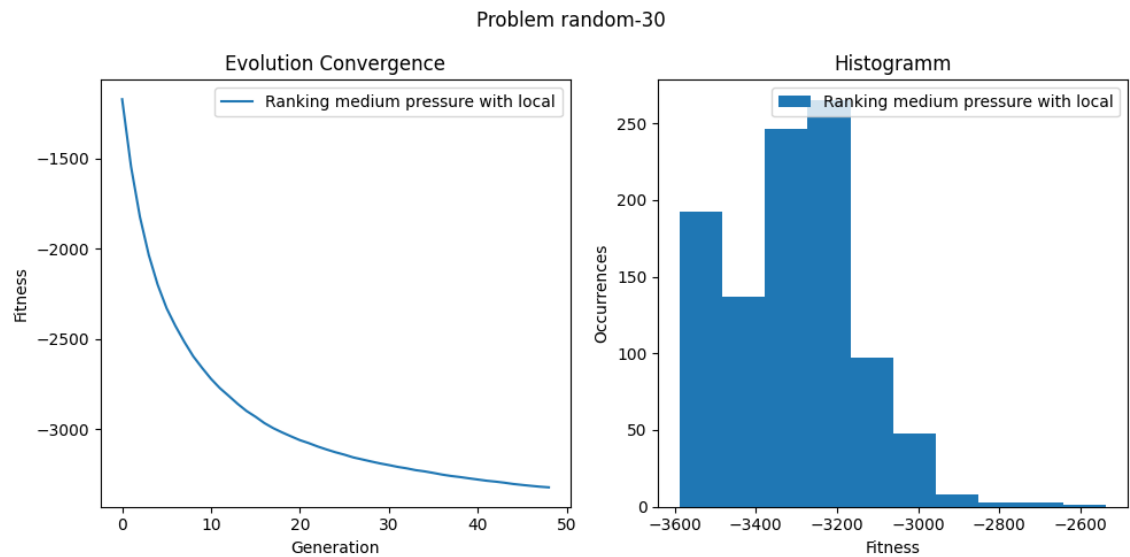
Tested Parameter Configuration: Ranking medium pressure with local

```

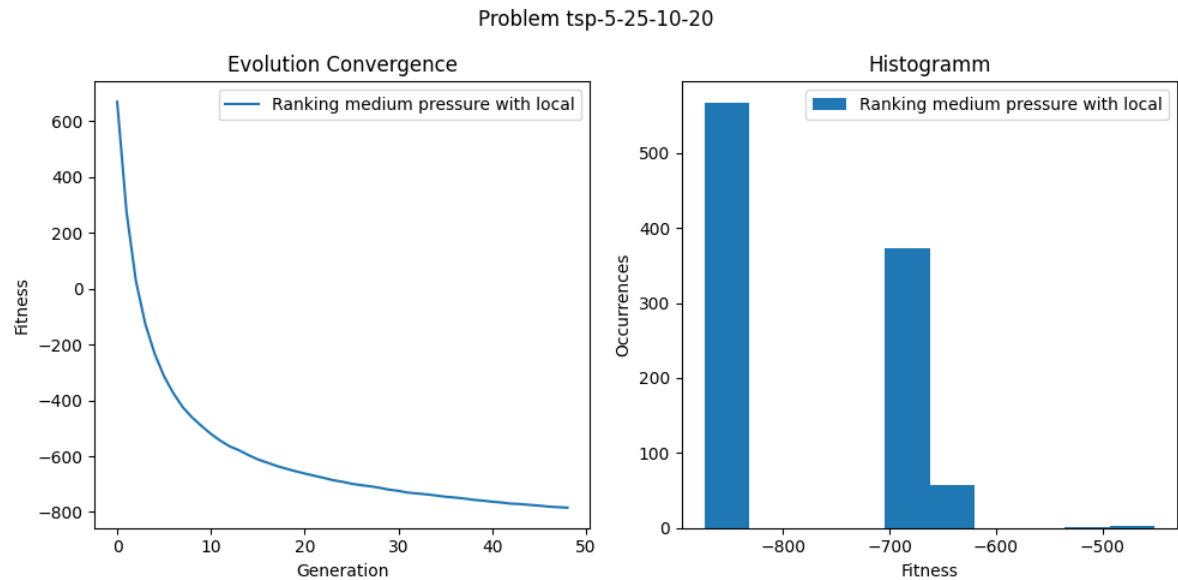
Population Size:      24
Generation Count:     48
Mutation Rate:        0.05
Crossover Rate:       1
Selection Pressure:    1.5
Optimization Rate:     1
Adjust Parameters:    False
Annealing Size:        5
Sub Problems:         [[]]
Init Strategy:         Random
Selection Strategy:    Ranking
Local Optimization:    LocalOptimization.SimAnnealingThermalRandom
Replacement Strategy:  Replacement.ParentAndOffSpring

```



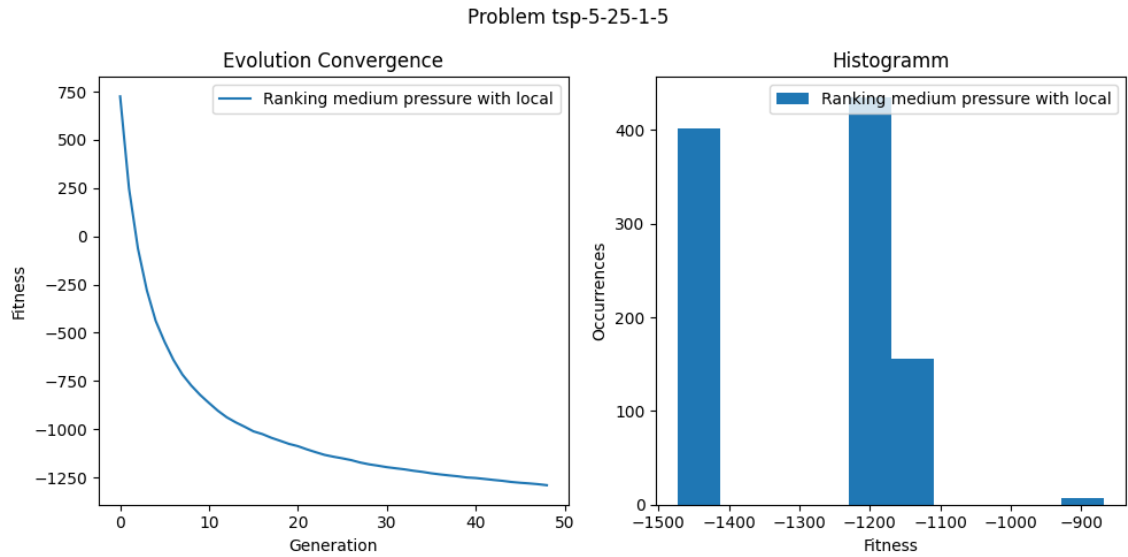


	Parameters	Accuracy	Average	Best Result	Worst
0	Ranking medium pressure with local	13.4	-3321.705	-3588	-2536



	Parameters	Accuracy	Average	Best Result	Worst
0	Ranking medium pressure with local	9.4	-784.331	-874	-451

	Parameters	Accuracy	Average	Best Result	Worst
0	Ranking medium pressure with local	10.9	-930.007	-1115	-585



	Parameters	Accuracy	Average	Best Result	Worst
0	Ranking medium pressure with local	6.6	-1290.601	-1473	-867

```
C:\Users\bmeins\Desktop\Master Arbeit\code\annealing\classic.py:27: RuntimeWarning: overflow encountered in exp
  x = update_weights @ np.array([x, 1 / (1 + np.exp(potential / temperature))])
```

In []: