The Traveling Tournament Problem

Exercise 2

Angelika Ferstl   Bernhard Mallinger (0707663)

January 17, 2012

Contents

1 Ant Colony Optimization

The ant colony optimization mimics the swarm intelligence of ants. The ants mark a path using pheromones while they are searching for food. A better path means higher intensity of the pheromons.

2 Pheromone model

For the TTP, there are several reasonable ways in which pheromones can be distributed to attributes of solutions. We identified several basic alternatives:

$n \times n$**-edge-matrix.** The entries (i, j) of this matrix would encode the desirability for a game against j after i. This is however very limited, i.e. it does not include information about the location of the game, which is most crucial for the value of objective function.

$2n \times 2n$**-edge-matrix.** This can be interpreted as an extension of the first variant, which distinguishes between home- and away-games, i.e. the range $[n + 1, 2n]$ represents games against this city abroad. Here, the problem still occurs that an away-game after a home-game can mean very different travelling costs depending on the location of the city with respect to the particular opponent.

Edge-matrix per city. To account for the different locations of the cities, one could consider the travelling order for every city as a distinct problem, therefore keeping a pheromone matrix for every city. The information can be expected to be more precise with at the expense of far higher construction costs. This approach can be combined with either one of the first two approaches.

Position-matrix. Instead of saving the edges, one could also save the absolute position of cities in the solution. Since the costs of a solution are composed of the single moves from one location to the next one, this alternative is not advisable.

In our implementation, we decided to use one $2n \times 2n$-edge-matrix in order to profit from synergies of applying the results of the tours for each city to one piece of data. The idea is that if many cities e.g. play an away-game against b after they played against a, the edge between those two is very likely to be a good solution attribute, and should therefore be used by as many cities as possible. This would not be possible in the same way if a matrix per city was used.

3 Construction heuristics

We implemented two different construction heuristics:

3.1 Greedy ants

This heuristic is based on the greedy algorithm we implemented in exercise 1: For every round, a city is chosen for game selection. Cities with few possible games are preferred. For the chosen city, all possible games are evaluated according to this objective function:

$$p_{ij} = \alpha * \eta_{ij} + \tau_{ij}$$

Among these, the minimum is selected.

The local information η_{ij} (distance from the current location to all other cities) is scaled into the range of $[0, 1]$ by dividing by the maximum distance in order to be able to make it compatible with the pheromone values. The factor α is used to reduce the influence of local information on the decision over time.

The pheromone values τ_{ij} are initialized by 1.0. Higher ranked ants (ants that are the most successful) are able to apply more pheromones and in our case three ants are allowed to apply them.

3.2 Randomized ants

This construction approach is more dynamic and also general than the first one. Instead of leveraging local information, the decisions are solely based upon the distribution of pheromones. The algorithm can therefore select solution attributes, that seem to be bad locally and those will be rewarded, if it leads to a good overall solution, which is not possible with greedy.

Since the search is purely random at the beginning, it takes far more iterations/ants than the greedy approach to find reasonable solutions, but this can be attributed to the fact that a far greater portion of the search space is considered.

In order to maximize the number of solutions that can be found, we also allow invalid solutions with respect to the repeaters- and number of home- and away-games-constraints in this approach. Those are punished by a fixed fine per violation so that nearly valid solutions are competitive against bad valid solutions, and solutions with an unreasonable number of violations are discarded.

4 Results

We used 500 iterations for every result and averaged them over 20 instances. The standard deviation (std) is to the left of the average cost result. We tried different numbers of ants and our best results were mostly found with 150 ants.

Ant without neighborhood - Randomized ants:

	NL4, f.sol. 8276		NL6, f.sol. 23978		NL8, f.sol. 41505		NL10, f.sol. 68691	
Ants	Costs	std	Costs	std	Costs	std	Costs	std
5	9072,25	571,23	29946,25	1923,33	52185,25	838,44	87186	1681
10	8729,25	367,05	27808,25	1451,74	50699,25	2594,40	86855,5	1350,5
20	8615,75	166,71	26901,75	380,58	49447,5	2849,26	87014,5	1416,5
50	8423,25	166,71	27025,25	620,41	46639,25	1205,16	87391,33	629,62
100	8519,5	192,50	27119,25	434,28	49371	1213,42	86610,66	1409,48
150	8519,5	192,50	26881,75	237,91	47826,5	848,79	84314,33	822,63
200	8423,25	166,71	26430,75	304,02	47860,25	1297,65	83207,66	940,41

Ant without neighborhood - Greedy ants:

	NL4, f.sol. 8276		NL6, f.sol. 23978		NL8, f.sol. 41505		NL10, f.sol. 68691	
Ants	Costs	std	Costs	std	Costs	std	Costs	std
5	8313,75	57,59	25188	444,85	43196,75	219,99	69703,5	597,5
10	8286,5	6,06	25005,25	211,15	42589	406,07	69119,5	180,5
25	8283	7,00	24885,25	243,22	42580,75	103,60	67693	227
50	8279,5	6,06	25139	358,60	42149,75	169,99	67744,5	118,5
100	8276	0	24946,25	308,61	41550,5	422,48	66908	121
150	8276	0	24923,00	190,16	41408	183,77	67409	191
200	8276	0	24801,25	448,83	41670,75	258,09	68237	216

Of course, we improved the results through local neighborhood search (change home/away pattern, rounds and teams). For the N4 test instance the best solution was always found and the average value of NL6 is close to the optimal solution. The greedy ants with local neighborhood search are able to find solutions for N8 and N10 not far from the optimum.

Ant with neighborhood, Randomized ants:

	NL4, f.sol. 8276		NL6, f.sol. 23978		NL8, f.sol. 41505		NL10, f.sol. 68691	
Ants	Costs	std	Costs	std	Costs	std	Costs	std
5	8276	0	24684	356,80	43230,5	556,47	73153	21
10	8276	0	24293,75	299,50	42814,75	650,81	74105,5	846,5
25	8276	0	24277,25	208,51	42458,25	1264,38	73687,5	149,5
50	8276	0	24367,75	311,85	42327	1194,70	72319,5	436,5
100	8276	0	24144,5	97,62	42074,5	925,54	71787	1047
150	8276	0	24143,25	114,85	41540	495,37	71239	900
200	8276	0	24155,5	109,18	41662,25	731,21	72585,5	930,5

Ant with neighborhood Greedy ants:

	NL4, f.sol. 8276		NL6, f.sol. 23978		NL8, f.sol. 41505		NL10, f.sol. 68691	
Ants	Costs	std	Costs	std	Costs	std	Costs	std
5	8276	0	24098	24,01	41322,25	281,03	66066	1728
10	8276	0	24239	177,89	41168	301,72	65595,5	858,5
25	8276	0	24333	266,25	41065,75	150,27	65816,5	1472,5
50	8276	0	24240	165,21	40960,25	112,26	64523	1972,43
100	8276	0	24283	360,27	41042,25	196,05	64807	655,50
150	8276	0	24155,5	109,18	40991,75	80,49	65477	309,27
200	8276	0	24443,25	266,35	40995,25	121,18	66411	2588,36