Algorithmics	Student information	Date	Number of session
	UO: 283428		6
	Surname: Triana Fernández		
	Name: Carlos		

Escuela de Ingeniería Informática
Universidad de Oviedo

Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Activity 1. Measurements

n	Time_BT	Time_BT_balancing	ZNCC_greedy	ZNNC_BT	ZNNC_BT_balancing
2	10	1	0,005348	0,005348	0
3	26	13	0,010732	0,025115	0,025115
4	66	40	0,002165	0,029852	0,022174
5	170	70	0,008006	0,038655	0,03866
6	568	349	0,035406	0,054626	0,05463
7	1555	998	0,033144	0,052461	0,05246
8	4653	2812	0,01064	0,053614	0,05361
9	13478	10743	0,021686	0,073122	0,07312
10	40710	30757	0,02789	0,075552	0,07555
11	127702	85213	0,0477	0,086866	0,086866
12	384395	301138	0,03231	0,089045	0,089045
13	1061921	868030	0,034626	0,097307	0,097307
14	3289608		0,042764	0,103836	

(On the greedy algorithm, n represents the number of times the algorithm is run)

(The balance factor on the Backtracking with balancing is n/3)

Activity 2. Questions

a) State the algorithm that provides better results and explain why.

The algorithm that provides better results is backtracking. It is the only algorithm that will find the best solution, as the greedy algorithm will find a solution, but it is not guaranteed to be a good one.

However, the greedy algorithm, due to having a significantly better complexity, will find a solution in less time.

Finally, as seen in the data obtained from the measurements, the most reasonable option is the Backtracking algorithm with a balance factor, because its lower time but same accuracy as the backtracking. This algorithm's solutions are the same as the ones obtained with regular backtracking, but avoiding calculating all solutions.

b) Which algorithm will you use for processing a realistic dataset with a million of images? Explain why.

In this case, I will consider that a backtracking algorithm wouldn't be a possible option with my resources, so I will use a greedy algorithm. In addition, due to the nature of the problem, where we don't require the perfect solution, a greedy algorithm has more advantages.

c) Determine the theoretical time complexity for backtracking (without balancing condition) and validate this analysis from the experimental results.

The overall complexity of the backtracking algorithm is factorial, considering that at each stage of the process, the length of the problem is reduced in one unit.

The values are unexpectedly low. If we take into account the values of n = 6 and n = 7 we have a time of 568 and 1555, but the derived value from 568 would be 568 * 7 = 3976 that is far from the result (calculating the value as (n2! / n1!) * t1).

d) Determine the advantage of including the balancing condition in terms of time for backtracking, does it affect the quality of the results?

The including of balancing conditions can dramatically lower the response time of the problem. Nevertheless, it doesn't change the complexity of the algorithm which remains factorial.

With respect to the quality of the results, it may not be affected due to the conditions and the results can be the same as the ones of regular backtracking. As seen in the measurements, the condition has no effect on the results, but significant lower times.