Catholic University of Louvain

Project 3 : Search

LINGI2365 - Constraint Programming

Auteurs :
Vanwelde Romain (3143-10-00)
Crochelet Martin (2236-10-00)

Groupe 7

Superviseurs :
Pr. Yves Deville
François Aubry

23 mars 2014

Table des matières

1 The Brussels airport problem

1.1 Explain the given model

1.2 Design 2 different variable and/or value ordering heuristics for this problem.

1.3 Which criteria are meaningful for comparing different search strategies ?

1.4 Based on your criteria, compare your heuristics with the labelFF heuristic by testing them on the instance on iCampus.

1.5 Consider the following strategy. . . . Give an example with three planes where this strategy is wrong

2 The Knapsack Problem

2.1 A Branch & Bound approach

2.1.1 Model the knapsack problem as Constraint Optimization

2.1.2 Describe your model in the report.

2.1.3 Design 3 different heuristics for variable selection.

2.1.4 Test your heuristics and the labelFF heuristic on the knapsack-A instances.

2.1.5 Present and discuss the results in your report

time :

	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10
s1	76	37	21	50	73	95	219	318	1098	2189
s2	15	10	9	14	20	27	63	124	213	349
s3	29	21	22	28	34	63	145	166	899	1476
ff	17	15	10	18	26	45	98	113	672	1034

#choices

	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10
s1	2151	1068	609	1488	2058	2668	6177	10206	36020	73265
s2	489	343	329	569	697	891	2309	3986	8218	12734
s3	1219	678	736	858	1016	2103	4877	5850	29973	49940
s4	538	468	339	617	860	1590	3756	4431	26737	41081

#fail

	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10
s1	4266	2098	1140	2940	3975	5298	12193	20306	71907	146472
s2	796	507	473	790	1186	1451	3792	7176	13027	21112
s3	1731	1120	1045	1454	1852	3667	8608	10247	55578	90988
s4	722	574	437	847	1149	1966	4781	5723	33206	54804

#propag :

	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10
s1	25586	12644	7115	16302	25829	36401	81296	129069	427978	819394
s2	3634	2197	1840	2451	4878	6845	15387	34646	50685	91129
s3	6250	5299	3798	6189	8609	17699	41002	48293	280535	391867
s4	6306	5371	3867	6210	8680	17772	41030	48393	280603	391948

2.2 Optimization over iterations

2.2.1 Model the knapsack problem as a Constraint Satisfaction Problem.

2.2.2 In order to implement the optimization over iterations . . .

2.2.3 Which of these points (i., ii., iii.) do you need to execute on which events ?

2.2.4 How do you modify the value of ub to be sure to find the optimal solution ?

2.2.5 Can you explain why we initialize ub with an upper bound instead of any other value ?

2.2.6 Experiment this program on the instances knapsack-A, -B.

2.2.7 Present and discuss the results in your report.

2.3 Optimization via divide and conquer

2.3.1 In order to implement the optimization via divide and conquer you will have to . . .

2.3.2 Which of these points (i., ii., iii., iv.) do you need to execute on which events ?

2.3.3 Experiment this version on the instances knapsack-A,-B,-C

2.3.4 Present and discuss the results in your report.