Heuristic Optimisation

Problem sheet 2

Brief solutions

1. Consider the following 6 dimensional SAT problem.

$$F(x_1, x_2, x_3, x_4, x_5, x_6) = (x_1 \bigvee \overline{x}_2 \bigvee x_3) \bigwedge (\overline{x}_2 \bigvee \overline{x}_6) \bigwedge (x_3 \bigvee x_6)$$

Enumerate the search space for this problem. Construct the truth table and find the solutions to the SAT problem, if there are any.

Solution

The search space has size $2^6 = 64$. The values for x_4 and x_5 are not used for calculating $F(x_1, x_2, x_3, x_4, x_5, x_6)$, so we only need to look at the values of the remaining variables.

For example, no matter what the values of x_4 and x_5 are, $x_1 = 0$, $x_2 = 0$, $x_3 = 0$, $x_6 = 1$ is a solution.

The truth table for the four variables (the X stands for "don't care" values):

x_1	x_2	x_3	x_6	$x_1 \bigvee \overline{x}_2 \bigvee x_3$	$\overline{x}_2 \bigvee \overline{x}_6$	$x_3 \bigvee x_6$	F
0	0	0	0	X	X	0	0
0	0	0	1	1	1	1	1
0	0	1	0	1	1	1	1
0	0	1	1	1	1	1	1
0	1	0	0	X	X	0	0
0	1	0	1	X	0	X	0
0	1	1	0	1	1	1	1
0	1	1	1	X	0	X	0
1	0	0	0	X	X	0	0
1	0	0	1	1	1	1	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	1
1	1	0	0	X	X	0	0
1	1	0	1	X	0	X	0
1	1	1	0	1	1	1	1
1	1	1	1	X	0	X	0

Note that for each solution shown in the truth table we have 4 different solutions which all have the given values for variables x_1, x_2, x_3, x_6 and have the 4 different combinations for x_4 and x_5.

2. Consider the following 4 dimensional SAT problem.

$$F(x_1, x_2, x_3, x_4) = (x_1 \bigvee \overline{x}_2 \bigvee x_3) \bigwedge (x_2 \bigvee \overline{x}_3) \bigwedge (x_3 \bigvee x_4) \bigwedge x_1$$

Apply the GSAT local search algorithm from starting point $x_1 = 0$, $x_2 = 0$, $x_3 = 0$, $x_4 = 0$. Can you find a solution if $MAX\_FLIPS = 2$?

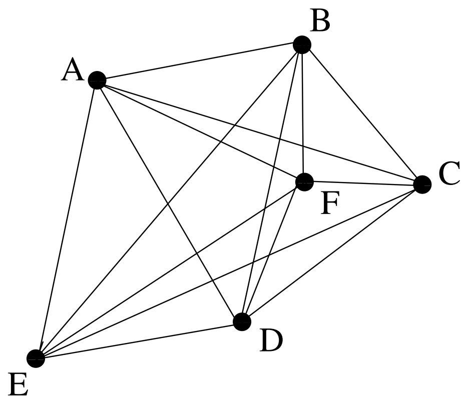
Solution

Flip	x_1	x_2	x_3	x_4	$x_1 \bigvee \overline{x}_2 \bigvee x_3$	$x_2 \bigvee \overline{x}_3$	$x_3 \bigvee x_4$	x_1	Nr. satisfied
0	0	0	0	0	1	1	0	0	2
1	1	0	0	0	1	1	0	1	3
	0	1	0	0	0	1	0	0	1
	0	0	1	0	1	0	1	0	2
	0	**0**	**0**	**1**	**1**	**1**	**1**	**0**	**3**
2	1	0	0	1	1	1	1	1	4

So, a solution could be found in two steps (repeating the flipping twice) from this starting point.

3. Consider a symmetric TSP problem with 6 cities A, B, C, D, E, F. The distances between cities are as follows:

City	City	Distance
A	B	1
A	C	3
A	D	3
A	E	4
A	F	2
B	C	2
B	D	4
B	E	5
B	F	1
C	D	2
C	E	3
C	F	1
D	E	1
D	F	2
E	F	2



(a) Apply the nearest neighbour heuristic optimisation method starting from city A and show the solution that you find.

(b) Apply the nearest neighbour heuristic optimisation method starting from city C and show the solution that you find.

(c) Explain whether differences can occur when applying the nearest neighbour method from different starting points.

(d) Apply a different greedy heuristic for solving this TSP problem. Is your solution better or worse than the ones found in (a) and (b)?

Solution

(a) The closest city to A is B with a distance of AB=1. Therefore, we go to B. From B we can go to C, D, E, or F. The closest of these is F with a distance of BF=1. Therefore, we go to F. From F we can go to C, D, or E. The closest of these is C with a distance of FC=1. Therefore, we go to C. From C we can go to D or E. The closest of these is D with a distance of CD=2. Therefore, we go to D. From D we can go to E only with a distance of DE=1. Finally, we complete our tour (A,B,F,C,D,E,A). The length of tour is AB+BF+FC+CD+DE+EA=1+1+1+2+1+4=10.

(b) The closest city to C is F with a distance of CF=1. Therefore, we go to F. From F we can go to A, B, D, or E. The closest of these is B with a distance of FB=1. Therefore, we go to B. From B we can go to A, D, or E. The closest of these is A with a distance of BA=1. Therefore, we go to A. From A we can go to D or E. The closest of these is D with a distance of AD=3. Therefore, we go to D. From D we can go to E only with a distance of DE=1. Finally, we complete our tour (C,F,B,A,D,E,C). The length of tour is CF+FB+BA+AD+DE+EC=1+1+1+3+1+3=10.

(c) Let us start the neighbour heuristic optimisation method at B. The closest city to B is A with a distance of BA=1. Therefore, we go to A. From A we can go to C, D, E, or F. The closest of these is F with a distance of AF=1. Therefore, we go to F. From F we can go to C, D, or E. The closest of these is C with a distance of FC=1. Therefore, we go to C. From C we can go to D or E. The closest of these is D with a distance of CD=2. Therefore, we go to D. From D we can go to E only with a distance of DE=1. Finally, we complete our tour (B,A,F,C,D,E,B). The length of tour is BA+AF+FC+CD+DE+EB=1+2+1+2+1+5=12. The length of these tour is larger than the ones found in (a) and (b). Hence, differences can occur when applying the nearest neighbour method from different starting points.

(d) We use the following heuristic: From all possible edges select the shortest one and add it to the tour. At the beginning all edges are available. The shortest edge length is 1. We choose one of the shortest edges: AB. All remaining edges with length 1 are available. We choose one: DE. All remaining edges with length 1 are available. We choose one: CF. There is only one edge with length 1 remaining: BF. We choose BF. B already belongs to two edges: AB and BF. Hence, no other edge is available which contains B. F already belongs to two edges: BF and CF. Hence, no other edge is available which contains F. Thus, the available edges are AD, AE, CD, CE. The shortest of these is CD=2. There is only one edge available AE which completes our tour (A,B,F,C,D,E,A) with length AB+BF+FC+CD+DE+EA=10. In fact this is the same tour as the one found in (a), hence its length is the same as the length of the tour found in (a), which is the same as the length of the tour found in (b). So, the solution is neither better nor worse than the ones found in (a) and (b), but equally as good as those.

4. Consider the 0/1 knapsack problem:

A set of N items, each having a value v_i and a weight w_i are given. There is a bag of finite capacity K (maximum weight it can hold). The problem is to fill the bag with items (without exceeding its capacity), so that the total value of items in the bag is maximised. You can either put an item in the bag as a whole or leave it out, it is not possible to use parts of an item.

(a) Describe a greedy algorithm for this problem.

(b) For the particular case:
$N = 5$
$v_1 = 55,\ w_1 = 50$
$v_2 = 50,\ w_2 = 24$
$v_3 = 32,\ w_3 = 30$
$v_4 = 24,\ w_4 = 20$
$v_5 = 20,\ w_5 = 10$
$K = 100$
would the greedy algorithm find the optimal solution?

Solution:

(a) The greedy step can be selecting the item with the *"best value for money"*, that is the item with maximum v_i/w_i and putting it in the bag if the bag can hold it. The stopping criterion: there are no more items that can fit in the bag, in the sense that if the selected item does not fit in the bag, then consider the next item in the decreasing order of v_i/w_i and put it in the bag if it fits.

(b) We have

$$\frac{v_2}{w_2} > 2 = \frac{v_5}{w_5} > \frac{v_4}{w_4} = 1.2 > 1.1 = \frac{v_1}{w_1} = 1.1 > \frac{v_3}{w_3}$$

Therefore, the greedy algorithm selects "items $2, 5, 4$ in the bag". The sequence $2, 5, 4$ is the order of selection of the items by the greedy algorithm, based on the decreasing order of v_i/w_i. The constraint for the total weight $W_{2,5,4} = w_2 + w_5 + w_4$ of this selection is satisfied because

$$W_{2,5,4} = w_2 + w_5 + w_4 = 24 + 20 + 10 = 54 < 100.$$

The next item selected by the greedy algorithm would be item 1. But this item cannot be selected because the total weight of the selection $2, 5, 4, 1$ would be

$$W_{2,5,4,1} = w_2 + w_5 + w_4 + w_1 = 24 + 20 + 10 + 50 = 104 > 100$$

which exceeds the capacity $K = 100$ of the bag. Hence, you skip it and consider the next item in the decreasing order of v_i/w_i, which is item 3. There are no more items to be considered, so you stop. The constraint for the total weight $W_{2,5,4,3} = w_2 + w_5 + w_4 + w_3$ of this solution is satisfied because

$$W_{2,5,4,3} = w_2 + w_5 + w_4 + w_3 = 24 + 20 + 10 + 30 = 84 < 100.$$

The total value of the solution found by the greedy algorithm is

$$V_{2,5,4,3} = v_2 + v_5 + v_4 + v_3 = 50 + 20 + 24 + 32 = 126.$$

This solution is not optimal because the solution "items $1, 2, 4$ in the bag" with a total weight

$$W_{1,2,4} = w_1 + w_2 + w_4 = 50 + 24 + 20 = 94 \leq 100.$$

has a higher total value

$$V_{1,2,4} = v_1 + v_2 + v_4 = 55 + 50 + 24 = 129.$$

5. Consider the following Boolean Satisfiability problem:

$$F(x_1, x_2, x_3, x_4, x_5) = (\bar{x}_1 \bigvee \bar{x}_2) \bigwedge (x_1 \bigvee \bar{x}_2 \bigvee \bar{x}_4) \bigwedge (x_1 \bigvee x_2 \bigvee \bar{x}_3) \bigwedge (\bar{x}_1 \bigvee \bar{x}_5).$$

Apply a greedy algorithm to solve this problem. Describe each step.

Solution: $x_1 = 0$ satisfies two clauses F_1 and F_4 and $x_1 = 1$ satisfies two clauses F_2 and F_3. It is a draw, choose for example $x_1 = 0$. $x_2 = 0$ satisfies one currently unsatisfied clause F_2 and $x_2 = 1$ satisfies one currently unsatisfied clause F_3. It is a draw, choose for example $x_2 = 1$. x_3 is not contained in the only currently unsatisfied clause F_2 and therefore has no influence on F_2. Therefore x_3 can take any values, for example choose $x_3 = 0$. $x_4 = 1$ does not satisfy the only currently unsatisfied clause F_2 and $x_4 = 0$ it satisfies it. Hence, $x_4 = 0$ satisfies more currently unsatisfied clauses than $x_4 = 1$. Therefore, choose $x_4 = 0$. All clauses are now satisfied and thus x_5 can be arbitrarily chosen. For example choose $x_5 = 0$. The found solution is

$$(x_1, x_2, x_3, x_4, x_5) = (0, 1, 0, 0, 0).$$