Artificial Intelligence 1
Lab 2

Roeland Lindhout (s2954524) & Younes Moustaghfir (s2909758)
AI2/AI3

May 18, 2016

1 Constraint Satisfaction Problems

Solving a small set of equations

The program took quite some time, and states, to calculate the variables. The only solution that the program found, was A = 3, B = 7 and C =2. This was found in 63254.

Market

We used three variables, each with a domain from 0 to 100. The three variables represented the number of fruits. Added to each other, they had to equal 100. The second constraint considered the prices: $A * 88 + B * 99 + C * 102 = 10000$. Wherein A represented the oranges, B the grapefruit and C the melons. The reason that they have to be equal to 10000, is that the prices are given in cents.

There were 5 solutions, in 10308 states. We checked the solutions, and they were all correct. The solutions were as follows:

```
### Solution 1 ###
A =    1
B =    62
C =    37

### Solution 2 ###
A =    4
B =    48
C =    48

### Solution 3 ###
A =    7
B =    34
C =    59

### Solution 4 ###
```

```
A =    10
B =    20
C =    70

### Solution 5 ###
A =    13
B =    6
C =    81
```

Chain of trivial equations

Without an extra constraint, many states were visited, and 100 solution werer found. The solutions were of course all the integers from 0 up to and including 99. With the constraint of A being 42, 2502 states were visited, and only 1 solution was found, namely all are 42. With the constraint of Z being 42, only 27 states were visited. The reason that this is so much lower, is that if $A = 42$, you have yet to determine all other variables, so you have to test all of them. When $Z = 42$, you can backtrack using the known final value, because each variable gets the value of the next variable.

Constraint graph

For this problem, there are 5 variables, namely A to E. Each of these variables has a domain from 1 up to and including 4. We had to introduce an extra variable, to be able to represent the constraint $C \neq D + 1$. This variable ranged from 2 to 4, because it is always bigger than 1, since 0 was not in the original domain. 5 (i.e. 4 + 1) also didn't have to be included, because it is impossible for C to be 5, so when D = 4, $C \neq D + 1$ is always true.

The problem was solved in 29 states, and 2 solutions were found:

```
### Solution 1 ###
A =    3
B =    3
C =    4
D =    2
E =    1
F =    3

### Solution 2 ###
A =    4
B =    4
C =    2
D =    3
E =    1
F =    4
```

Cryptarithmetic puzzles

For the SEND+MORE=MONEY-problem, we had to define 7 variables with a domain ranging from 0 to 9 and 4 variables ranging from 0 to 1. The first 7 variables were all the letters of those words. In this case that were: S, E, N, D, M, O, R, Y. SEND and MORE are like a sum, so carry-over variables are needed. In this case we named them: X1 to X4, because the total length of the words SEND and MORE is 4, therefore there are 4 places where a carry-over could take place. The constraints we chose were:

```
alldiff(S,E,N,D,M,O,R,Y);
D + E = Y + 10 *X1;
X1 + N + R = E + 10 * X2;
X2 + E + O = N + 10 * X3;
X3 + S + M = O + 10 * X4;
X4 = M;
M = 1;
```

Most of the constraints are quite logical, but we chose the value 1 for M, because the values of S and M cannot be higher than 20(because they are both maximally 9). Therefore M cannot be higher than 1. M can also not be 0, because numbers do not start with a zero.

For the UN+UN+NEUF = ONZE-problem, we had to make a small adjustment compared to the code for the first problem described in this subsection, because we did not have two, but three words, so it became possible that the carry-over could also be 2. As a heuristic, we figured that X3, which was the last carry-over, could not be bigger than 1, because at that point, we no longer had 3 letters. We did not need a fourth carry-over, because there is no further addition in the final step. There is only one solution. Namely:

```
### Solution 1 ###
U =    8
N =    1
E =    9
F =    7
O =    2
Z =    4
X1 =   0
X2 =   2
X3 =   1
```

2 Source code