CS 5800: Algorithms K.Lieberherr, V. Pavlu

Due: April. 14, 2014

Homework Module 11

1 Submission Rules

`http://www.ccs.neu.edu/home/lieber/courses/algorithms/cs5800/sp14/homeworks/submission-rules.pdf`

2 Problems

1. (25 pts) Exercise 24.2-2. Shortest Paths for DAGs
2. (25 pts) Exercise 24.3-2. Dijkstra's algorithm
3. (20 pts) Exercise 25.1-4. Shortest Paths and Matrix Multiplication.
4. (20 points) Plan your Wedding. (Concluding hw 5)

   `http://www.ccs.neu.edu/home/lieber/courses/algorithms/cs5800/sp14/labs/wedding-organization.html`

   We are looking for your algorithm that is guaranteed to find the maximum seating arrangements in polynomial time. You are welcome to test your algorithm by doing debates but in this case we only grade the final result: your algorithm and your argument for correctness.

5. Approximation of Maximum Generalized Satisfiability Problems using Randomization (40 points)

   Consider Maximum Generalized Satisfiability problems using one relation R given by its truth table. An instance S consists of n Boolean variables used in m constraints all using relation R. We call such an instance S an R-instance. The goal is to find an assignment to the Boolean variables that maximizes the fraction of satisfied constraints in a given R-instance. For most R this maximization problem is NP-hard.

   Notation:

   $$assignments(S) =$$

   the set of all Boolean assignments to variables of the constraints S.

   $$fsat(S, J) =$$

   the fraction of satisfied constraints in constraints S under assignment J.

Consider the following claim:

$$ConstraintThresholdClaim = \forall R \in \text{BooleanRelations } \exists t \in [0,1] : CTC(R,t)$$

where

$$CTC(R,t) =$$
$$\forall S \in R\text{−instances } \exists J \in assignments(S) : fsat(S,J) \geq t$$

and

$$\exists S_0 \in R - instances \ \forall J_0 \in assignments(S_0) : fsat(S_0, J_0) < t + 10^{-3}$$

The t value gives an approximation bound for an approximation algorithm that tries to come close to the maximum.

For each of the following relations R find $t = f(R)$ so that $CTC(R, f(R))$ holds.

Example of R-instance with 4 constraints in 4 variables x1,x2,x3,x4:

```
R(x1,x2,x3)
R(x1,x2,x4)
R(x1,x3,x4)
R(x2,x3,x4)
```

Let's assume that R=R1 from below. The R-instance is contradictory and we cannot satisfy all 4 constraints. The assignment J=(1,1,1,0) satisfies 3 of the 4 constraints.

Hint: try a perfectly unbiased coin ($p = 1/2$) first and then try to improve on it by making the coin biased. p is the probability that any of the variables is set to 1 (true).

You need a formal property of random variables (C.21), called linearity of expectation, which is explained on page 1198 of CLRS.

What to turn in: For each relation R the value $f(R)$. (Optional, extra credit (40 points)) A polynomial-time algorithm that satisfies the fraction $f(R)$ in any R-instance.

Below are truth tables defining the Boolean relations that we consider. A truth table for a relation with arity n has 2^n rows and $n+1$ columns where the last one is the result column.

```
R1
    Result
000 0
001 0
010 0
011 1
100 0
```

101 1
110 1
111 0

R2

000 0
001 0
010 1
011 1
100 0
101 1
110 1
111 0

R3

000 0
001 1
010 1
011 1
100 1
101 1
110 1
111 0

R4
000 0
001 1
010 0
011 0
100 0
101 0
110 0
111 0

R5
0000 0
0001 1
0010 1
0011 0
0100 1

```
0101 0
0110 0
0111 0
1000 1
1001 0
1010 0
1011 0
1100 0
1101 0
1110 0
1111 0
```

6. Release and share all your algorithms. During the semester there were a few instances where you kept your algorithms secret. Now is the time to share them in your groups.