GANDALF 2018: Programming task 2

Read the following excerpt from an unpublished draft of scientific paper

1 Introduction

Given a finite alphabet Σ of letters, a *word* w is a finite sequence of letters. We denote the set of all finite words over Σ by Σ^*. For a word w, we define $w[i]$ as the i-th letter of w, and we define $w[i,j]$ as the subword $w[i]w[i+1]\dots w[j]$ of w. We use the same notation for other sequences defined later on. By $|w|$ we denote the length of w.

A *(non-deterministic) finite automaton* (NFA) is a tuple $(\Sigma, Q, Q_0, F, \delta)$ consisting of an input alphabet Σ , a finite set of states Q, a set of initial states $Q_0 \subseteq Q$, a set of final states F, and a finite transition relation $\delta \subseteq Q \times \Sigma \times Q$.

Weighted automata. A *weighted automaton* is a finite automaton whose transitions are labeled by rational numbers called *weights*. Formally, a weighted automaton is a tuple $(\Sigma, Q, Q_0, F, \delta, C)$, where the first five elements are as in the finite automata, and $C \colon \delta \to \mathbb{Q}$ is a function that defines *weights* of transitions. Figure 1 contains an example of a weighted automaton.

A *run* π of an automaton $\mathcal{A}$ on a word w is a sequence of states $\pi[0]\pi[1]\dots$ such that $\pi[0]$ is an initial state and for each i we have $(\pi[i-1], w[i], \pi[i]) \in \delta$. A run π of length k is *accepting* if and only if the last state $\pi[k]$ belongs to the set of accepting states F. Every run π of an automaton $\mathcal{A}$ on a word w defines a sequence of weights of successive transitions of $\mathcal{A}$ as follows. Let $(C(\pi))[i]$ be the weight of the i-th transition, i.e., $C(\pi[i-1], w[i], \pi[i])$. Then, $C(\pi) = (C(\pi)[i])_{1 \leq i \leq |w|}$. A *value functions* f is a function that assigns real numbers to sequences of rational numbers. The value $f(\pi)$ of the run π is defined as $f(C(\pi))$. We consider here only one value function, $\textsc{Sum}$, defined as $\textsc{Sum}(\pi) = \sum_{i=1}^{|C(\pi)|} (C(\pi))[i]$.

The value of a (non-empty) word w assigned by the automaton $\mathcal{A}$, denoted by $\mathcal{L}_{\mathcal{A}}(w)$, is the infimum of the set of values of all accepting runs on w. The value of a word that has no (accepting) runs is infinite.

1.1 Probabilistic semantics

A *terminating* Markov chain $\mathcal{M}$ is a tuple $\langle \Sigma, S, s_0, E, T \rangle$, where Σ, S and s_0 are as usual, $E \colon S \times (\Sigma \cup \{\epsilon\}) \times S \mapsto [0,1]$ is the edge probability function, such that if $E(s,a,t)$, then $a = \epsilon$ if and only if $t \in T$, and for every $s \in S$ we have $\sum_{a \in \Sigma \cup \{\epsilon\}, s' \in S} E(s,a,s') = 1$, and T is a set of terminating states such that the probability of reaching a terminating state from any state s is positive. Notice that the only ϵ-transitions in a terminating Markov chain are those that lead to a terminating state.

The probability of a finite word u w.r.t. $\mathcal{M}$, denoted $\mathbb{P}_{\mathcal{M}}(u)$, is the sum of probabilities of paths from s_0 labeled by u such that the only terminating state on this path is the last one.

The distribution of infinite words is *uniform* if it is given by a terminating Markov chain with one regular state and one terminating state; it loops over any letter in the non-terminating state with probability $\frac{1}{|\Sigma|+1}$ or go to the terminating state over ϵ with probability $\frac{1}{|\Sigma|+1}$.

Automata as random variables. A weighted automaton defines the function $\mathcal{L}_{\mathcal{A}}(w) \colon \Sigma^* \mapsto \mathbb{R}$ that assigns values to words. The measurealibity of this function can be proved in the standard manner. Thus, this function can be interpreted as random variables w.r.t. the probabilistic space we consider. Hence, for a given automaton $\mathcal{A}$ and a Markov chain $\mathcal{M}$, we consider the following quantities:

$\mathbb{E}_{\mathcal{M}}(\mathcal{A})$ — the expected value of the random variable defined by $\mathcal{A}$ w.r.t. the probability measure defined by $\mathcal{M}$.
$\mathbb{D}_{\mathcal{M},\mathcal{A}}(\lambda) = \mathbb{P}_{\mathcal{M}}(\{w \mid \mathcal{L}_{\mathcal{A}}(w) \leq \lambda\})$ — the (cumulative) distribution function of the random variable defined by $\mathcal{A}$ w.r.t. the probability measure defined by $\mathcal{M}$.

1.2 Computational questions

We consider the following basic computational questions:

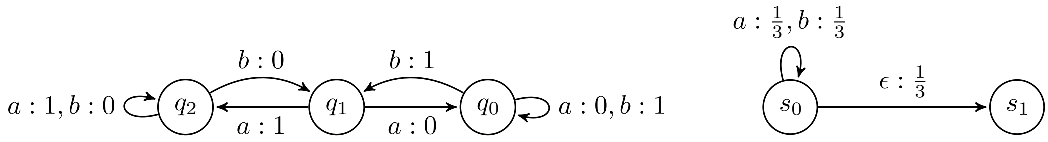

Figure 1: The automaton $\mathcal{A} = \{\{a,b\},\{q_0,q_1,q_2\},\{q_0\},\emptyset,\delta,C\}$, where $\delta = \{(q_0,a,q_0),(q_0,b,q_0),(q_0,b,q_1),(q_1,a,q_0),(q_1,a,q_2),(q_2,a,q_1),(q_2,a,q_2),(q_2,b,q_2)\}$ and C such that $C(q_0,b,q_0) = C(q_2,a,q_2) = C(q_0,b,q_1) = C(q_1,a,q_0) = 1$ and for all other inputs the value of C is 0 (left) and the Markov chain $\mathcal{M} = \{\{a,b\},\{s_0,s_t\},\{s_0\},E\}$ where E always returns $\frac{1}{3}$ (right).

The expected value question: Given an f-automaton $\mathcal{A}$ and a terminating Markov chain $\mathcal{M}$, compute $\mathbb{E}_{\mathcal{M}}(\mathcal{A})$.
The distribution question: Given an f-automaton $\mathcal{A}$, a terminating Markov chain $\mathcal{M}$ and a threshold λ, compute $\mathbb{D}_{\mathcal{M},\mathcal{A}}(\lambda)$.
The approximate expected value question: Given an f-automaton $\mathcal{A}$, a terminating Markov chain $\mathcal{M}$, $\epsilon > 0$, compute a number η such that $|\eta - \mathbb{E}_{\mathcal{M}}(\mathcal{A})| \leq \epsilon$.
The approximate distribution question: Given an f-automaton $\mathcal{A}$, a terminating Markov chain $\mathcal{M}$, a threshold λ and $\epsilon > 0$ compute a number $\eta \in [\mathbb{D}_{\mathcal{M},\mathcal{A}}(\lambda - \epsilon), \mathbb{D}_{\mathcal{M},\mathcal{A}}(\lambda + \epsilon)]$.

2 Tasks

Assume $\Sigma = \{a, b\}$. Write a program that solves:

Problem 1. (4 points) The approximate expected value question for Sum-automata. (If you wish, you can assume that the distribution is uniform for -1 point and that the automaton is deterministic for -1 point; with both assumptions, you can get at most 2 points).

Problem 2. (3 points) The approximate distribution question for Sum-automata.

Problem 3. (4 points*) The expected value question for deterministic Sum-automata. *Hint: this perhaps requires some knowledge on weighted Markov chains.*

2.1 IO specification

The input consists of:

- The specification of an automaton, that start with a line containing two positive integers: n m where n denotes the number of states, and then has m lines containing transitions, each of the form
  $s\ l\ w\ s'$
  where $s, s' \in \{0, \ldots, n-1\}$ are the states, $l \in \{a, b\}$ is the letter and $w \in \mathbb{Q}$ is the weight. We assume that 0 is the initial state and that all the states are final.
- The specification of a terminating Markov chain, which consists of a line containing two positive integers:
  n' m'
  and m' lines of the form
  $s\ l\ p\ s'$
  where $s, s' \in \{0, \ldots, n'-1\}$ are the states and $l \in \{a, b, -\}$ is the letter and $p \in [0, 1]$ is the probability. We assume that state 0 is the only initial state and 1 is the only terminating state.
- A *query*, which is a single line that may be E e (which stands for "find the expected value" given $\epsilon = e$) or D x e (which stands for "compute the distribution value for $\lambda = x$ and $\epsilon = e$").

  Every rational number is given as "numerator/denominator", e.g., 1/3.

  The output should be a single rational number.

2.2 Example input

```
3 8
0 a 0/1 0
0 b 1/1 0
0 b 1/1 1
1 a 0/1 0
1 a 1/1 2
2 b 0/1 1
2 a 1/1 2
2 b 0/1 2
2 3
0 a 1/3 0
0 b 1/3 0
0 - 1/3 1
E 1/4
```