Institut für Theoretische Physik
der Universität zu Köln

Rochus Klesse
Laurens Ligthart , Felipe Montealegre-Mora

Quantum Information Theory – Sheet 4

Wintersemester 2021/22

Webpage: *http://www.thp.uni-koeln.de/∼rk/qit_22.html/*

Submission of solutions as pdf-file until Thursday, June 2, 12 pm, to *ligthart.exams[at]gmail.com*

12. RSA-decoding

1+1+4=6 points

An RSA-encrypted message E can be easily decoded when one knows its order r modulo n, where n together with an integer e co-prime to $\varphi(n)$ constitutes the public key (e, n). The original message M (co-prime to n), encoded as $E = M^e \mod n$, can then be decoded from E by

$$M = E^{d'} \mod n\,,$$

where d' is the inverse of e modulo r. In the lecture this was shown by using the following fact:

For e co-prime $\varphi(n)$ and x co-prime n, the order of x^e modulo n equals the order of x modulo n

In order to prove this first show:

a) Any m satisfying $x^m = 1 \mod n$ is an integer multiple of the order of x modulo n.

b) For x co-prime n, the order of x modulo n divides $\varphi(n)$

then, building on **a)** and **b)**, eventually show:

c) For e co-prime $\varphi(n)$ and x co-prime n, the order r of x^e modulo n equals the order $\tilde{r}$ of x modulo n.

Hints: $\varphi(n)$ denotes Euler's φ-function; **a)**: proof by contradiction; **b)**: use Euler's theorem.

13. Grover's algorithm

6 points

Show that if the number of solutions is $t = N/4$, then Grover's algorithm always finds a solution with certainty after just one query. How many queries would a classical algorithm need to find a solution with certainty if $t = N/4$? And if we allow the classical algorithm an error probability of $1/10$?

14. Searching for the nimimum

6 points

N numbers $x_1, x_2, \ldots, x_N$ are stored in an unsorted manner in a *quantum* data base. Give a quantum algorithm that finds the smallest element x_i within an *expected* number of $O(\sqrt{N})$ data base queries. How many queries would need your algorithm in the worst case?

Hint: Assume that a Grover-like search on the data base can be performed and that this search finds one of t items within N unsortet elements with an *expected* number of $O(\sqrt{\frac{N}{t}})$ queries, and this also when the number t is *unknown*.

15. Turing machines and the Halting-function

2+2+2=6 points

a) What is the effect of the following Turing machine on a general binary input word x ?

$$M = (\{s_0, s_1, s_2, s_e\}, \{0, 1\}, \{0, 1, \Box\}, \delta, s_0, \Box, \{s_e\}) \quad (1)$$

with transition function

$$\begin{aligned} \delta: \quad s_0, 0 &\rightarrow s_0, 0, R \\ s_0, 1 &\rightarrow s_0, 1, R \\ s_0, \Box &\rightarrow s_1, 1, R \\ s_1, \Box &\rightarrow s_2, 1, L \\ s_2, 0 &\rightarrow s_2, 0, L \\ s_2, 1 &\rightarrow s_2, 1, L \\ s_2, \Box &\rightarrow s_e, \Box, R \end{aligned}$$

b) Design a Turing machine that computes $f(x) = x \bmod 2$.

c) The incomputability of the Halting-function

$$h(w) = \begin{cases} 1 & \text{Turing machine } M_w \text{ holds on input } w \\ 0 & \text{Turing machine } M_w \text{ does not hold on input } w \end{cases}$$

can be proven by contradiction. To this end it is assumed that a Turing machine M exists that on any input w holds after some finite time with output $h(w)$, i.e.

$$s_0 w \vdash^* s_e h(w) ,$$

where s_0 and s_e are initial and final states of M. In the proof this machine M needs to be modified into a Turing machine M' with the property that on input w

1. M' holds (e.g. with output 0) when M on input w would output 0 (i.e. $h(w) = 0$)

2. M' will never hold when M on input w would output 1 (i.e. $h(w) = 1$)

Explicitly construct M' starting from the machine M. To be specific assume that M has internal sates $s_0, \ldots s_n, s_e$, binary alphabet $\Sigma = \{0, 1\}$, working alphabet $\Gamma = \{0, 1, \Box\}$ and transition rules δ. Add internal states and correspondingly extend the transition rules such that the result is the machine M' with the above properties.