Quantum Information Theory, Spring 2019

Problem Set 15 **due May 27, 2019**

1. *(4 points)* **Hermitian-preserving maps:** Recall that a superoperator $\Phi \in \mathrm{T}(\mathcal{X},\mathcal{Y})$ is *Hermitian preserving* if $\Phi(X) \in \mathrm{Herm}(\mathcal{Y})$, for all $X \in \mathrm{Herm}(\mathcal{X})$. Show that the following are equivalent:

   (a) $\Phi \in \mathrm{T}(\mathcal{X},\mathcal{Y})$ is Hermitian preserving,
   (b) $(\Phi(X))^* = \Phi(X^*)$, for all $X \in \mathrm{L}(\mathcal{X})$,
   (c) $\Phi^* \in \mathrm{T}(\mathcal{Y},\mathcal{X})$ is Hermitian preserving,
   (d) $J(\Phi) \in \mathrm{Herm}(\mathcal{Y} \otimes \mathcal{X})$.

2. *(4 points)* **Finding adjoint:** Let $\Phi \in \mathrm{T}(\mathcal{X},\mathcal{Y})$ and $\Psi \in \mathrm{T}(\mathcal{X},\mathcal{Z})$ be Hermitian-preserving maps. Define a superoperator $\Xi \in \mathrm{T}(\mathcal{X}\oplus\mathcal{Z},\mathcal{Y}\oplus\mathcal{Z})$ as follows:

$$\Xi\begin{pmatrix} X & \cdot \\ \cdot & Z \end{pmatrix} = \begin{pmatrix} \Phi(X) & 0 \\ 0 & \Psi(X)+Z \end{pmatrix},$$

   for all $X \in \mathrm{L}(\mathcal{X})$ and $Z \in \mathrm{L}(\mathcal{Z})$. Find the adjoint superoperator $\Xi^* \in \mathrm{T}(\mathcal{Y}\oplus\mathcal{Z},\mathcal{X}\oplus\mathcal{Z})$.

3. *(4 points)* **SDP for the 1-norm:** Recall the semidefinite program from Problem 2 in the exercise set, which was specified in terms of some matrix $K \in \mathrm{L}(\mathcal{X},\mathcal{Y})$:

   Primal problem

$$\begin{aligned} \text{maximize:}\quad & \mathrm{Re}(\langle K, Z\rangle) \\ \text{subject to:}\quad & \begin{pmatrix} I_{\mathsf{X}} & Z^* \\ Z & I_{\mathsf{Y}} \end{pmatrix} \geq 0, \\ & Z \in \mathrm{L}(\mathcal{X},\mathcal{Y}). \end{aligned}$$

   Dual problem

$$\begin{aligned} \text{minimize:}\quad & \frac{1}{2}\mathrm{Tr}(X) + \frac{1}{2}\mathrm{Tr}(Y) \\ \text{subject to:}\quad & \begin{pmatrix} X & -K^* \\ -K & Y \end{pmatrix} \geq 0, \\ & X \in \mathrm{Pos}(\mathcal{X}), \\ & Y \in \mathrm{Pos}(\mathcal{Y}). \end{aligned}$$

   Let α and β denote the optimal primal and dual values. Show that $\alpha = \beta = \|K\|_1$.

4. *(4 points)* **Practice:** The goal of this problem is to find measurements that discriminate quantum states from a given ensemble. The files `R1.txt`, `R2.txt`, `R3.txt` contain *real* 5×5 density matrices. Let us denote these states by ρ_1, ρ_2, ρ_3.

   (a) Use Helstrom's bound to determine the optimal success probability for discriminating the states from the ensemble

$$\left(\frac{1}{2}\rho_1, \frac{1}{2}\rho_2\right).$$

(b) Write down an SDP for this state discrimination task. What numeric value for the probability of success does it produce when you run it?

(c) Consider the ensemble of three states, each given with equal probability:

$$\left(\frac{1}{3}\rho_1, \frac{1}{3}\rho_2, \frac{1}{3}\rho_3\right).$$

What is the optimal probability of success for discriminating these states? Formulate this problem as an SDP and solve it numerically.

(d) Solve the same three-state discrimination problem using the "pretty good measurement". Compare its success probability to the optimal measurement from the previous part.

Example: *This example illustrates how SDPs can be solved numerically. Let*

$$P = |+\rangle\langle+| = \frac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

and consider the following simple problem:

$$\begin{aligned} \text{maximize:} \quad & \langle P, X\rangle \\ \text{subject to:} \quad & I \geq X \geq 0. \end{aligned}$$

The optimal value of this SDP is clearly 1 *and the corresponding solution is of the form*

$$X = |+\rangle\langle+| + c|-\rangle\langle-|,$$

for some $c \in [0,1]$. *See the code for solving this SDP here: Python and Mathematica.*

Python hint: *You can solve SDPs using packages like PICOS and CVXPY. Let us focus on PICOS. To install it, run*

```
pip install picos --user
```

PICOS uses the following syntax:

- `(P|X)` stands for $\langle P, X\rangle$,
- `X >> 0` stands for $X \geq 0$,
- `'I'` stands for the identity matrix I.

You can create a new SDP problem `p` by writing

```
p = picos.Problem()
```

To specify a variable `x` that is a symmetric matrix `d` $\times$ `d`, write

```
x = p.add_variable('x', (2,2), 'symmetric')
```

You can add the semidefinite constraint $\mathtt{x} \geq 0$ as follows:

```
p.add_constraint(x >> 0)
```

To specify the objective function as maximization of $\langle a, x \rangle$, write

```
p.set_objective('max', (a|x))
```

To solve the problem, use

```
p.solve(verbose = 0, solver = 'cvxopt')
```

Mathematica hint: *Older versions of Mathematica unfortunately do not have built-in support for solving SDPs, and external packages (such as NCAlgebra) are unreliable and difficult to use. Luckily, this has changed since version 12 which was released last month. If you do not have access to it through your university, you can use the web interface provided by Wolfram Cloud.*

In Mathematica 12, SDPs are solved using the function

`SemidefiniteOptimization`$[f, cons, vars]$

where f is a linear function, $cons$ is a list of constraints, and $vars$ is a list of scalar variables appearing in the function and constraints. In addition to regular equalities and inequalities as possible constraints, you can also include semidefinite constraints:

- $X \geq 0$: `VectorGreaterEqual[{X,0},"SemidefiniteCone",d]`
- $X \leq I$: `VectorLessEqual[{X,IdentityMatrix[d]},"SemidefiniteCone",d]`

where `d` *denotes the dimension of the matrices involved. Note that you have to explicitly parametrize the matrix entries of all positive semidefinite matrices you want to optimize over. Since Mathematica does not support SDPs with complex numbers, all involved positive semidefinite matrices must actually be real and symmetric. You can create a symmetric* $\mathtt{d} \times \mathtt{d}$ *matrix with entries* $\mathtt{M}_{\mathtt{i,j}}$ *using the following function:*

```
SymmetricMatrix[M_,d_]:=Table[Subscript[M,Sequence@@Sort@{i,j}],{i,1,d},{j,1,d}];
```