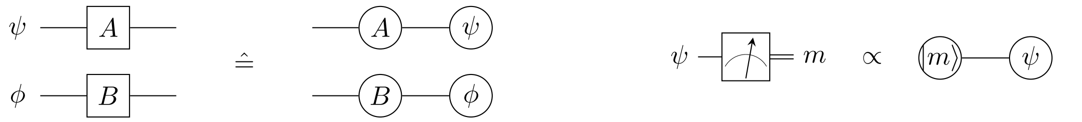
Christian B. Mendl, Richard M. Milbradt due: 17 June 2022 08:15

Tutorial 7 (Quantum teleportation as tensor network)
Mathematically, a qubit ψ is a vector in $\mathbb{C}^2$ with norm 1. Basis states in quantum (bra-ket) notation are written as $|0\rangle = \left(\begin{smallmatrix}1\\0\end{smallmatrix}\right)$ and $|1\rangle = \left(\begin{smallmatrix}0\\1\end{smallmatrix}\right)$, such that a qubit can be represented as $\psi = \alpha\,|0\rangle + \beta\,|1\rangle$ with $\alpha, \beta \in \mathbb{C}$.

Quantum circuit diagrams are read from left to right, and the wires can be interpreted as tensor legs. As example, translating to the notation used in the lecture, with $\psi, \phi \in \mathbb{C}^2$, matrices $A, B \in \mathbb{C}^{2\times 2}$ and $m \in \{0, 1\}$:

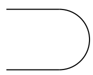


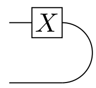
Here the "input" of the left two diagrams is the outer product $\psi \circ \phi$, or equivalently (when interpreted as vector) the Kronecker product $\psi \otimes \phi$. For basis states one writes $|ab\rangle = |a\rangle \otimes |b\rangle$. On the right, m is the "measurement outcome", and $\propto$ means "proportional to".

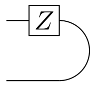
The *Pauli matrices* are both Hermitian and unitary, and given by

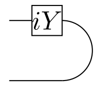
$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

For the following, we also define the so-called *Bell states* β_{ab}, which can be regarded as vectorizations of the identity and Pauli matrices:

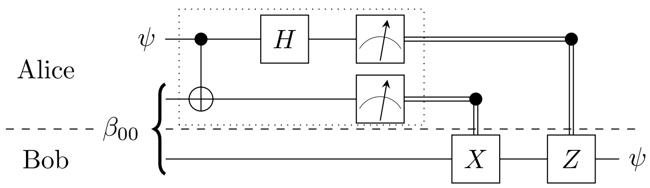
$$\beta_{00} = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right) \mathrel{\hat{=}} \tfrac{1}{\sqrt{2}}$$ 

$$\beta_{01} = \frac{1}{\sqrt{2}}\left(|01\rangle + |10\rangle\right) \mathrel{\hat{=}} \tfrac{1}{\sqrt{2}}$$ 

$$\beta_{10} = \frac{1}{\sqrt{2}}\left(|00\rangle - |11\rangle\right) \mathrel{\hat{=}} \tfrac{1}{\sqrt{2}}$$ 

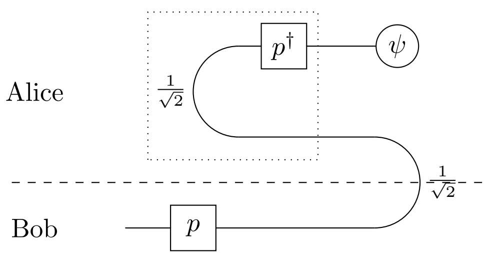
$$\beta_{11} = \frac{1}{\sqrt{2}}\left(|01\rangle - |10\rangle\right) \mathrel{\hat{=}} \tfrac{1}{\sqrt{2}}$$ 

Quantum teleportation involves two parties called Alice and Bob, and allows Alice to transmit the data (the coefficients α and β) of a qubit ψ to Bob. Initially they share the Bell state β_{00}. Alice performs local operations on ψ and her half of β_{00}, then transmits two bits of classical information to Bob, who can then recover the original qubit ψ. The protocol is summarized by the following circuit:



(a) Describe the components and individual steps of the quantum teleportation circuit.

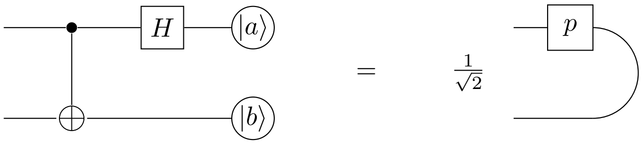
(b) According to Exercise 7.1, the dotted region in the circuit can be interpreted as projection onto one of the Bell states (depending on the measurement outcomes). With this information, map the circuit to the following tensor network diagram (read from right to left), where p is the identity matrix, X, iY or Z:



What is the final state at Bob according to this diagram?

(c) A tensor network is deterministic. However, the measurement outcome and the correction unitary is not. How can this be reconciled?

Exercise 7.1 (Bell circuit, part 2)
Generalize Exercise 4.1 by proofing the following equivalence for all $a, b \in \{0, 1\}$, where p is the identity matrix, X, iY or Z (depending on a and b). The circuit thus generates one of the Bell states. Hint: See also Exercise 4.2(d).



Exercise 7.2 (Implementation of Poisson's equation using a Tucker format Ansatz)
Our goal is to implement the ALS algorithm described in Tutorial 6, to solve Poisson's equation using a Tucker format approximation for ϕ:

$$-\Delta\phi = \mathtt{b} \quad \text{on } \Omega = [0,1]^3,$$
$$\phi|_{\partial\Omega} = 0$$

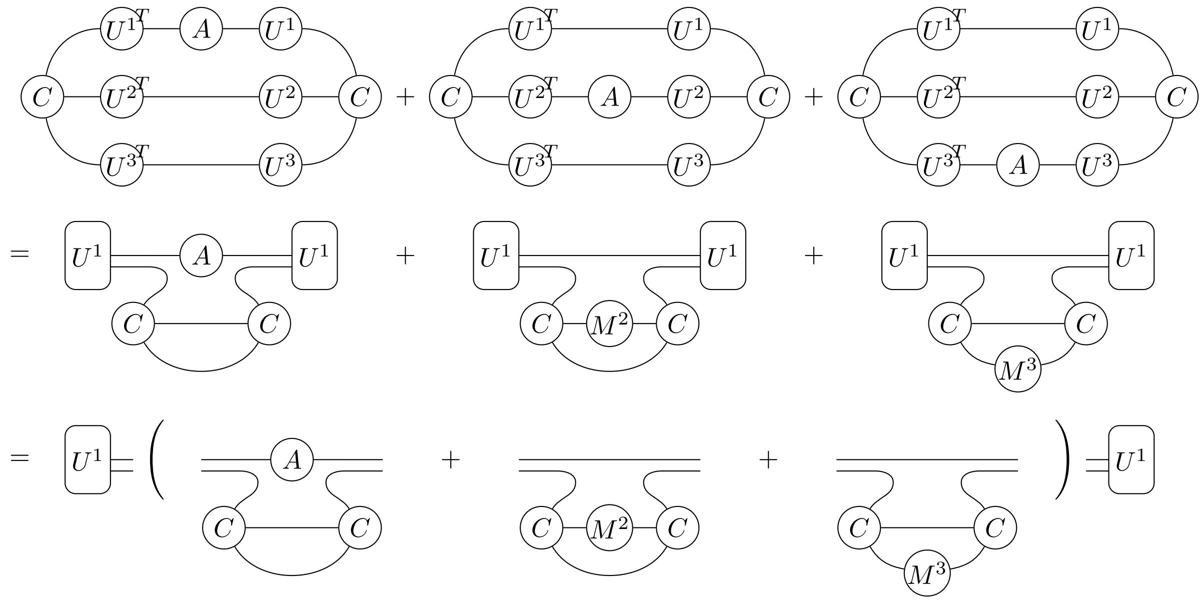
The Moodle page contains a template for this exercise, with placeholders for the following steps:

(a) Implement the finite difference approximation of $-\frac{\mathrm{d}^2}{\mathrm{d}x^2}$ on $[0, 1]$ with zero boundary conditions, by assembling the following matrix A in the function `fd_second_derivative_zero_boundary(n)`, given $n \in \mathbb{N}$ and corresponding $h = \frac{1}{n}$:

$$A = \frac{1}{h^2}\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & & \ddots & \\ & & & -1 & 2 \end{pmatrix} \in \mathbb{R}^{(n-1)\times(n-1)}$$

Hint: `np.diag(..., k)` can be used for constructing an off-diagonal matrix.

The ALS algorithm requires to express $\langle \phi, L\phi \rangle$ as quadratic function of each matrix U^j (interpreted as vector). We following diagrams illustrate how to construct this quadratic form for U^1. We use that U^2 and U^3 are isometries, i.e., $(U^2)^T U^2 = I$ and $(U^3)^T U^3 = I$, but do not exploit this property for U^1, since optimizing with respect to U^1 will result in a general matrix at first. We can then perform a QR decomposition, $U^1 = QR$, replace U^1 by Q and absorb R into C to ensure that U^1 remains an isometry. For conciseness, we define $M^j = (U^j)^T A U^j$ here:



In summary, the diagrams realize $\langle \phi, L\phi \rangle = \langle \mathfrak{u}^1, K^1 \mathfrak{u}^1 \rangle$ with $\mathfrak{u}^1 = \mathrm{vec}(U^1)$ and K^1 the matrix representation of the terms in the round brackets. The approach works analogously for U^2 and U^3.

(b) Construct each matrix K^j by implementing the function
`quadratic_form_tucker_isometry(A, phi: TuckerTensor, j)`.

(c) Finally, run the Jupyter notebook using your implementations. This should result in a Tucker format approximation of the solution to Poisson's equation for $\mathtt{b} : \Omega \to \mathbb{R}$, $\mathtt{b}(x, y, z) = \sin(3\pi(x + y + z))$, with a relative error around 10^{-4} compared to the numerically exact reference solution.