Create New Quantum Gates

by José Luis Gómez-Muñoz

http://homepage.cem.itesm.mx/lgomez/quantum/
jose.luis.gomez@itesm.mx

Introduction

This is a tutorial on the use of Quantum‘Computing’ Mathematica add-on to create new quantum gates.

Load the Package

First load the Quantum‘Computing’ package. Write:
Needs["Quantum`Computing`"]
then press at the same time the keys ≈-Û to evaluate. Mathematica will load the package.

A New Quantum Gate of One Qubit without Explicit Dirac Expression

Use the command SetQuantumGate to specify that a name (ga in this example) is to be consider a Quantum Gate. The first argument is the name of the new quantum gate, and the second argument is the number of input/output qubits of this new gate:
Press the keys \[ESC\]qg[ESC] in order to obtain the quantum gate template \(\Box\), press the \[TAB\] key to move between the empty place holders \(\Box\) and write the gate name (\(ga\) in this example, it is the name that was previously defined as a quantum gate) and also write the qubit number or label (3 in this example):

\[
\begin{align*}
g_a_3 \\
g_a_3
\end{align*}
\]

The new quantum gate that was defined above (with the command SetQuantumGate) can be included in the plot of quantum circuits:

\[
\text{QuantumPlot}[g_a_3]
\]

The new quantum gate cannot be used with more neither less qubits than the number that was specified in SetQuantumGate, therefore next command gives an error because \(ga\) cannot be used as a two-qubits gate, and the calculation is aborted:

\[
\begin{align*}
g_{a_3,4} \\
g_{a_3,4}
\end{align*}
\]

SetQuantumGate::nqb:

Quantum gate \(ga\) was called with 2 argument(s):{3,4}. It must have a number \(na\) of arguments such that 1 <= \(na\) <= 1

\$Aborted

On the other hand, the notation \(g_{a_{q1,q2}}\) with curly brackets in the subscript, is used for two gates of one qubit each. Therefore next input is accepted without errors. Press [ESC]qqg[ESC] for the "two gates of one qubit" template \(\Box_{\{,\}}\)

\[
\begin{align*}
g_{a_{3,4}} \\
g_{a_{3,4}}
\end{align*}
\]

We can use the two gates of one qubit in a quantum circuit:
QuantumEvaluate transforms the notation $g_{a_{3,4}}$ into the notation $g_{a_3} \cdot g_{a_4}$. Both mean the same: two gates of one qubit.

Use the \texttt{\textbackslash esc}qn\texttt{\textbackslash esc} template in order to easily write a large number of gates:

Notice that is very different the interpretation of $g_{a_5}$ from $g_{a_5}$ (it is very different \texttt{\textbackslash esc}qg\texttt{\textbackslash esc} from \texttt{\textbackslash esc}qn\texttt{\textbackslash esc}).
This is a "controlled" version of the new gate. Press [ESC]\{gate\}[ESC] for the \texttt{C^{(g)}} \{ \square \} template.
This is the matrix representation of the "controlled" version of the new gate. Press [ESC]cgate[ESC] for the $C^{[\widehat{2}]}$ template:

$$\begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}$$

This is the truth table of the "controlled" version of the new gate. Press [ESC]cgate[ESC] for the $C^{[\widehat{2}]}$ template:

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$0_1$, $0_2$</td>
</tr>
<tr>
<td>1</td>
<td>$0_1$, $1_2$</td>
</tr>
<tr>
<td>2</td>
<td>$1_1$, $0_2$</td>
</tr>
<tr>
<td>3</td>
<td>$1_1$, $1_2$</td>
</tr>
</tbody>
</table>

Notice that in the following circuit the three gates $g_a$ are not in the same column:
Notice that in the following circuit the three gates $g_a$ are not in the same column:

We can plot the three gates in the same column using the \texttt{\textbackslash placer\textbackslash template}, which can be entered by pressing the keys \texttt{\[ESC\]qggg[ESC]} (one q because they are one-qubit gates, three g because there are three gates):
QuantumEvaluate transforms $g_{a\{1,2,3\} > 0}$ into $g_a_1 \cdot g_a_2 \cdot g_a_3$. Both expressions have the same meaning, but the first one makes QuantumPlot use the same column for the three gates, as was shown above.

The "quantum register" notation [ESC]qn[ESC] gives more flexibility in the naming of the qubits:

The notation $\otimes_{j=1}^{3} g_{a_j}$ is transformed to $g_a_1 \cdot g_a_2 \cdot g_a_3$ without having to use QuantumEvaluate:
The expression $gb$ is a quantum gate of 1 arguments (qubits).

Here we can see the gate applied to the qubit $\hat{5}$:

$$\text{QuantumEvaluate}[gb]$$_{\hat{5}}$$

If the gate is not inside QuantumEvaluate, then it will not be transformed to its Dirac expression:
The new quantum gate that was defined above can be included in the plot of quantum circuits:

\[
\text{QuantumPlot}[gb_5]
\]

This is the truth table of the new quantum gate that was defined above:

\[
\text{QuantumTableForm}[gb_5]
\]

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(0_5) + (\frac{1}{2}) (0_5) (+) (\frac{1}{2}) (1_5) (\rangle)</td>
</tr>
<tr>
<td>1</td>
<td>(1_5) (-) (\frac{1}{2}) (0_5) (+) (\frac{1}{2}) (\sqrt{3}) (1_5) (\rangle)</td>
</tr>
</tbody>
</table>

This is the matrix representation of the new quantum gate that was defined above:

\[
\text{QuantumMatrixForm}[gb_5]
\]

\[
\begin{pmatrix}
\frac{\sqrt{2}}{2} & -\frac{1}{2} \\
\frac{1}{2} & \frac{\sqrt{3}}{2}
\end{pmatrix}
\]

This is the new gate in terms of Pauli operators:

\[
\text{PauliExpand}[gb_5]
\]

\[
\frac{1}{2} \sqrt{3} \sigma_{o,i} - \frac{1}{2} \sigma_{s,i}
\]

**A New Quantum Gate of Two Qubits With Explicit Dirac Expression**

The third (optional) argument of SetQuantumGate is the function that will be applied to the qubit(s) when the gate is inside the QuantumEvaluate command. The second argument (the number 2) indicates that this quantum gate takes as input two qubits, therefore the third argument must be a function of two qubits:

\[
\text{SetQuantumGate}[gc, 2, \text{Function}[(q1, q2), (0_\downarrow, 0_\uparrow) \cdot (0_\downarrow, 0_\uparrow) - (0_\downarrow, 1_\uparrow) \cdot (0_\uparrow, 1_\uparrow) + (1_\downarrow, 1_\uparrow) \cdot (1_\downarrow, 0_\uparrow) - (1_\downarrow, 0_\uparrow) \cdot (1_\downarrow, 1_\uparrow)]]
\]

The expression gc is a quantum gate of 2 arguments (qubits)
The new quantum gate cannot be used with more neither less qubits than the number that was specified in SetQuantumGate, therefore next command gives an error because gc cannot be used as a one-qubit gate, and the calculation is aborted:

\[
\text{gc}_5
\]

SetQuantumGate::nqb:
Quantum gate gc was called with 1 argument(s):\{9\}. It must have a number na of arguments such that \(2 \leq na \leq 2\)

\$Aborted

The new quantum gate gc was defined as a two-qubit gate. Press [ESC]qgg[ESC] for the two-qubits-one-gate template:

\[
\text{gc}_{5, 9}
\]

\[
\text{gc}^{5}_{5, 9}
\]

The commands above (at the top of this section) defined gc as a quantum gate with a specific Dirac expression. Here we can see the gate applied to the qubits \(5, 9\):

\[
\text{QuantumEvaluate}[\text{gc}^{5}_{5, 9}]
\]

\[
| 0_5, 0_9 \rangle \cdot \langle 0_5, 0_9 | - | 0_5, 1_9 \rangle \cdot \langle 0_5, 1_9 | + | 1_5, 1_9 \rangle \cdot \langle 1_5, 1_9 | - | 1_5, 0_9 \rangle \cdot \langle 1_5, 1_9 |
\]

This is the truth table of the new quantum gate that was defined above:

\[
\text{QuantumTableForm}[\text{gc}^{5}_{5, 9}]
\]

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(</td>
</tr>
<tr>
<td>1</td>
<td>(</td>
</tr>
<tr>
<td>2</td>
<td>(</td>
</tr>
<tr>
<td>3</td>
<td>(</td>
</tr>
</tbody>
</table>

The new quantum gate that was defined above can be included in the plot of quantum circuits:
The gate is separated in two blocks with the default qubit order. A vertical line is used to indicate that the two blocks represent a single two-qubits gate.

The option QubitList can be used to specify another order for the qubits, so that the gate gc can be plotted as a single block:
A New Quantum Gate With a Number of Qubits Inside a Range

The second argument of `SetQuantumGate` can be a list of two integers, specifying that the number of qubits for this gate must be within the interval spanned by those integers. If such kind gate is defined with the third (optional) argument of `SetQuantumGate`, then it is expected that the third argument will be a Function using the standard `Mathematica` notation for an arbitrary sequence of arguments: `##` (SlotSequence).

The expression `ge` is a quantum gate with a number `na` of arguments (qubits) such that `2 ≤ na ≤ 4`.

The quantum gate `ge` that was defined above cannot be used with only one qubit, therefore we obtain an error message and the calculation is aborted if we try to use it with only one qubit:

```
SetQuantumGate::nqb:
Quantum gate ge was called with 1 argument(s): 1. It must have a number na of arguments such that 2 <= na <= 4
```

The quantum gate `ge` can be used with 2, 3 or 4 qubits. Press [ESC]qqg[ESC] for the template:
In order to write the quantum gate with four qubits, first write with three \[\text{ESC}\]qqqg[\text{ESC}] \(\hat{e}_1, \hat{e}_2, \hat{e}_3\), then place the cursor to the right of the last qubit, and write another qubit pressing [\text{ESC}]qb[\text{ESC}] \(\hat{e}_4\).

\[\text{QuantumPlot} \left[ \text{ge}_{1,2} \cdot \text{ge}_{1,3,4} \right] \]

This is the truth table of \(\text{ge}_{1,2,3}\). This command can take several seconds in your computer to evaluate.

\[\text{QuantumTableForm} \left[ \text{ge}_{1,2,3} \right] \]

This is the representation of the gate in terms of Pauli operators. This calculation can take several seconds in your computer:
PauliExpand\[\text{ge}_{1,2,3}\]

\[\sigma_{0,1} \cdot \sigma_{0,2} \cdot \sigma_{0,3} + \frac{i}{\sqrt{2}} \sigma_{x,1} \cdot \sigma_{x,2} \cdot \sigma_{x,3} + \frac{2}{\sqrt{2}} \sigma_{x,1} \cdot \sigma_{x,2} \cdot \sigma_{x,3} - \frac{2}{\sqrt{2}} \sigma_{x,1} \cdot \sigma_{x,2} \cdot \sigma_{x,3}\]

\[2 \sqrt{2} \quad 2 \sqrt{2} \quad 2 \sqrt{2} \quad 2 \sqrt{2}\]

A New Quantum Gate With a Parameter

Nested \text{Function}[] commands can be used in order to define a parametric quantum gate, as shown in the example below. The outer \text{Function}[] handles the dependence on the qubits, while the inner \text{Function}[] handles the dependence on the parameter:

\text{SetQuantumGate}\[\text{rot}, 1,\]
\text{Function}\[(\text{q1}),\]
\text{Function}\[(\theta),\]
\text{Cos}[\theta] \cdot \text{0}_{\text{q1}} \cdot \text{0}_{\text{q1}} - \text{Sin}[\theta] \cdot \text{0}_{\text{q1}} \cdot \text{1}_{\text{q1}} + \text{Sin}[\theta] \cdot \text{1}_{\text{q1}} \cdot \text{0}_{\text{q1}} + \text{Cos}[\theta] \cdot \text{1}_{\text{q1}} \cdot \text{1}_{\text{q1}}\]

The expression rot is a quantum gate of 1 arguments (qubits).

The "parametric quantum gate" template $\text{pqg}[]$ can be entered by pressing the keys [ESC]pqg[ESC]. This is the truth table of the parametric quantum gate that was defined above, when the parameter is $\pi/6$ and it is applied to the qubit with label $\hat{3}$:

\text{QuantumTableForm}\[\text{rot}_{\hat{3}}[\pi/6]\]

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0$</td>
<td>$</td>
</tr>
<tr>
<td>$1$</td>
<td>$</td>
</tr>
</tbody>
</table>

The "parametric quantum gate" template $\text{pqg}[]$ can be entered by pressing the keys [ESC]pqg[ESC]. This is the truth table of the parametric quantum gate that was defined above, when the parameter is $\pi/12$ and it is applied to the qubit with label $\hat{3}$:

\text{QuantumTableForm}\[\text{rot}_{\hat{3}}[\pi/12]\]

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0$</td>
<td>$\left(\frac{\sqrt{2}}{2} + \frac{1}{2} \sqrt{2}\right)</td>
</tr>
<tr>
<td>$1$</td>
<td>$\left(\frac{\sqrt{2}}{2} - \frac{1}{2} \sqrt{2}\right)</td>
</tr>
</tbody>
</table>
The "parametric quantum gate" template \( c_\theta \) can be entered by pressing the keys [ESC]pqg[ESC]. This is the truth table of the parametric quantum gate that was defined above, when the parameter is \( \pi/4 \) and it is applied to the qubit with label \( 3 \):

\[
\text{QuantumTableForm}[\text{rot}_3[\pi/4]]
\]

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>(</td>
<td>0\rangle )</td>
</tr>
<tr>
<td>(</td>
<td>1\rangle )</td>
</tr>
</tbody>
</table>

The parametric quantum gate that was defined above can be included in quantum circuits:

\[
\text{QuantumPlot}[\text{rot}_3[\pi/4]]
\]

This is a "controlled version" of the new parametric quantum gate. Press [ESC]cgate[ESC] for the \( c^{(\dagger)}[\cdot] \) template, and [ESC]pqg[ESC] for the \( c_\theta \) template:

\[
\text{QuantumPlot}[c^{(\dagger)}[\text{rot}_3[\pi/4]]]
\]

This is the truth table for a "controlled version" of the new parametric quantum gate. Press [ESC]cgate[ESC] for the \( c^{(\dagger)}[\cdot] \) template, and [ESC]pqg[ESC] for the \( c_\theta \) template:
This is a "tensor power" of the parametric quantum gate. Notice that each gate in the answer is applied to a different qubit.

Press [ESC]pow[ESC] for the \((\ominus)^n\) template, and [ESC]pqg[ESC] for the \(I_0^{\ominus}\) template:

\[
\left(\text{rot}_{1}\left[\frac{\pi}{4}\right]\right)^{3}
\]

\[
\text{rot}_{1}\left[\frac{\pi}{4}\right] \cdot \text{rot}_{2}\left[\frac{\pi}{4}\right] \cdot \text{rot}_{3}\left[\frac{\pi}{4}\right]
\]

This is the truth table of a "tensor power" of the parametric quantum gate. Notice that there are three different qubits in the answer. Press [ESC]pow[ESC] for the \((\ominus)^n\) template, and [ESC]pqg[ESC] for the \(I_0^{\ominus}\) template:

\[
\text{QuantumTableForm}\left[\left(\text{rot}_{1}\left[\frac{\pi}{4}\right]\right)^{3}\right]
\]

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(</td>
</tr>
<tr>
<td>1</td>
<td>(</td>
</tr>
<tr>
<td>2</td>
<td>(</td>
</tr>
<tr>
<td>3</td>
<td>(</td>
</tr>
<tr>
<td>4</td>
<td>(</td>
</tr>
<tr>
<td>5</td>
<td>(</td>
</tr>
<tr>
<td>6</td>
<td>(</td>
</tr>
<tr>
<td>7</td>
<td>(</td>
</tr>
</tbody>
</table>

Compare the following truth table with the previous one. The difference is that below we have a "normal power" [ESC]po[ESC]; it means the repeated application of the gate to the same qubit:
This is a tensor product of an expression that includes the new parametric gate. Press \[\text{ESC}tprod\text{ESC}\] for the \(\otimes\) template, \[\text{ESC}pqg\text{ESC}\] for the \(\Box\) template, \[\text{ESC}hg\text{ESC}\] for the \(\hat{\text{H}}\) template and \[\text{ESC}on\text{ESC}\] for the centerdot \(\cdot\) symbol:

\[
\bigotimes_{j=1}^{3} \left( \text{rot}_{j+1} \left[ \frac{\pi}{4} \right] \cdot \hat{\text{H}}_j \right)
\]

\[
\text{rot}_{2} \left[ \frac{\pi}{4} \right] \cdot \hat{\text{H}}_1 \cdot \text{rot}_{3} \left[ \frac{\pi}{4} \right] \cdot \hat{\text{H}}_2 \cdot \text{rot}_{4} \left[ \frac{\pi}{4} \right] \cdot \hat{\text{H}}_3
\]

This is the quantum circuit for the expression. Notice that the three parametric gates are not in the same column:

The circuit looks more symmetrical with the option QuantumGateShifting → False, however the three parametric gates are not in the same column:
Finally a version of the circuit where the three parametric gates are in the same column. Notice that the tensor product template is not used anymore. Instead, press the keys [ESC]pqggg[ESC] for the the \( \phi \) template and [ESC]hggg[ESC] for the \( \hat{\mathcal{H}}(,\hat{\mathcal{H}},\hat{\mathcal{H}}) \) template:

These two notations represent the same set of gates, the difference is whether the gates will or will not be plotted in the same column in a quantum circuit:

This is the Dirac expression for our new parametric quantum gate, with parameter \( \pi/6 \):
This is the matrix representation of our new parametric quantum gate, with parameter \( \pi/6 \):

\[
\begin{pmatrix}
\frac{\sqrt{3}}{2} & -\frac{1}{2} \\
\frac{1}{2} & \frac{\sqrt{3}}{2}
\end{pmatrix}
\]

This is the matrix representation of the hermitian conjugate of our new parametric quantum gate, with parameter \( \pi/6 \). Press \([ESC]\)her\([ESC]\) for the \( (\pi)^\dagger \) template:

\[
\begin{pmatrix}
\frac{\sqrt{3}}{2} & \frac{1}{2} \\
-\frac{1}{2} & \frac{\sqrt{3}}{2}
\end{pmatrix}
\]

This is the Pauli expansion of our new parametric quantum gate, with parameter \( \pi/6 \):

\[
\frac{1}{2} \sqrt{3} \sigma_{0,3} - \frac{1}{2} i \sigma_{y,3}
\]

If we try to get the expansion for an arbitrary value of the parameter, in this case \( t \), we get an error, because if \( t \) is a complex number, then gate would be nonunitary:

\[\text{PauliExpand::nonunit: PauliExpand can only expand unitary operators}\]

The expression \( \text{rot}_3[t] \) is or might be nonunitary for some complex values of its parameters.

\[\$\text{Aborted}\]

A simple way to indicate that the parameter is always real is to write \( \text{Re}[t] \) instead of \( t \). We obtain the correct expansion without any error:

\[\text{PauliExpand[\text{rot}_3[\text{Re}[t]]]}\]

\[
\cos[\text{Re}[t]] \sigma_{0,3} - i \sin[\text{Re}[t]] \sigma_{y,3}
\]