August 9, 2024

Contents

0.1 Redfield Issue check "Analytically"

Since there seems to be an issue with the Bloch-Redfield Solver in qutip and my not so good implementation of the time dependent redfield, here's a quick sympy check to make sure it's not the solvers. By solving the equation for the SYK(2) model symbolically. The Hamiltonian is given by

$$H = \begin{bmatrix} -a & 0 & 0 & 0 \\ 0 & a & 0 & 0 \\ 0 & 0 & a & 0 \\ 0 & 0 & 0 & -a \end{bmatrix}$$

Here I define the coupling operator q that make things break for Bloch-Redfield on the SYK model

$$q = \begin{bmatrix} 0 & b_0 - ib_1 & b_2 - ib_3 & 0 \\ b_0 + ib_1 & 0 & 0 & -b_2 + ib_3 \\ b_2 + ib_3 & 0 & 0 & b_0 - ib_1 \\ 0 & -b_2 - ib_3 & b_0 + ib_1 & 0 \end{bmatrix}$$

I then get the eigenvalues and eigenvectors to obtain the jump operators (this steps are hidden in the pdf)

Jump operator checks

The jump operators must satisfy

$$[H, A(\omega)] = -\omega A(\omega) \tag{1}$$

$$[H, A^\dagger(\omega)A(\omega)] = 0 \tag{2}$$

$$\sum_w A(\omega) = A \tag{3}$$

There's a check below hidden in the pdf

Constucting the Differential equations

We now construct the differential equations from the GKLS form of the bloch Redfield generator

$$\dot{\rho_S^I(t)t} = \sum_{\omega,\omega',\alpha,\beta} \gamma_{\beta,\alpha}(\omega,\omega') \left(S_\alpha(\omega')\rho_S^I(t)S_\beta^\dagger(\omega) - \frac{\{S_\beta^\dagger(\omega)S_\alpha(\omega'), \rho_S^I(t)\}}{2} \right) + i \sum_{\omega,\omega',\alpha,\beta} S_{\beta,\alpha}(\omega,\omega') \left[\rho_S^I(t), S_\beta^\dagger(\omega)S_\alpha(\omega')\right]$$

I solve in the interaction picture generally, I did the same in my numerics so it should not be an issue (I rotate in the end). By neglecting Lambshift as in the numerics

$$\dot{\rho_S^I(t)t} = \sum_{\omega,\omega',\alpha,\beta} \gamma_{\beta,\alpha}(\omega,\omega') \left(S_\alpha(\omega')\rho_S^I(t)S_\beta^\dagger(\omega) - \frac{\{S_\beta^\dagger(\omega)S_\alpha(\omega'), \rho_S^I(t)\}}{2} \right) \tag{4}$$

As the sum goes on (ω, ω') pairs I construct all combinations

```
ws = list(jumps.keys())
combinations = list(itertools.product(ws, ws))
combinations
```

```
[(2*a, 2*a), (2*a, -2*a), ( -2*a, 2*a), ( -2*a, -2*a)]
```

I get the GKLS form of each of those combinations, as a dictionary

Then I construct the generator by multiplying the appropiate coefficient to each of the GKLS from matrices. Now for the coefficients we have

$$\Gamma_{\alpha,\beta}(\omega,t) = \int_0^t ds e^{i\omega s} \langle B_\alpha^\dagger(t) B_\beta^\dagger(t-s)\rangle_B \tag{5}$$

For convenience we also define

$$\Gamma_{\alpha,\beta}(\omega,\omega',t) = e^{i(\omega'-\omega)t}\int_0^t ds e^{i\omega s} \langle B_\alpha^\dagger(t) B_\beta(t-s)\rangle_B = e^{i(\omega'-\omega)t}\Gamma_{\alpha,\beta}(\omega,t) \tag{6}$$

Since I mainly care about bloch-redfield I make $t\to\infty$ (in the integral) so

$$\Gamma_{\alpha,\beta}(\omega,\omega',t) = e^{i(\omega'-\omega)t}\int_0^\infty ds e^{i\omega s} \langle B(s) B(0)\rangle_B = e^{i(\omega'-\omega)t}\Gamma_{\alpha,\beta}(\omega) \tag{7}$$

Where $\Gamma_{\alpha,\beta}(\omega)$ is the power spectrum

$\Gamma(w) = \frac{2\gamma\lambda^2 w\left(1+\frac{1}{e^{\frac{w}{T}}-1}\right)}{\gamma^2w^2+\left(-w^2+w_0^2\right)^2}$

Next we simply vectorize the density matrix and construct the system of ODES

$\frac{d}{dt}\rho_1(t) = 4ac_1\left(c_2\rho_{10}(t)\overline{c_3} + c_2\rho_6(t)\overline{c_2} + c_3\rho_{11}(t)\overline{c_3} + c_3\rho_7(t)\overline{c_2}\right)$

$\frac{d}{dt}\rho_2(t) = c_0\rho_2(t)$

$\frac{d}{dt}\rho_3(t) = c_0\rho_3(t)$

$\frac{d}{dt}\rho_4(t) = 4ac_1\left(-\rho_{10}(t)\overline{c_3}^2 + \rho_{11}(t)\overline{c_2c_3} - \rho_6(t)\overline{c_2c_3} + \rho_7(t)\overline{c_2}^2\right)$

$\frac{d}{dt}\rho_5(t) = ac_1\left(4c_2^2\rho_2(t) + 4c_2c_3\rho_3(t) - 4c_2\rho_{14}(t)\overline{c_3} - 4c_3\rho_{15}(t)\overline{c_3} - 2.0\left(c_2\overline{c_2} + c_3\overline{c_3}\right)\rho_5(t)e^{4iat}\right)e^{-4iat}$

$\frac{d}{dt}\rho_6(t) = 2c_0\rho_6(t)$

$\frac{d}{dt}\rho_7(t) = 2c_0\rho_7(t)$

$\frac{d}{dt}\rho_8(t) = ac_1\left(-4c_2\rho_2(t)\overline{c_3} + 4c_2\rho_3(t)\overline{c_2} - 2.0\left(c_2\overline{c_2} + c_3\overline{c_3}\right)\rho_8(t)e^{4iat} + 4\rho_{14}(t)\overline{c_3}^2 - 4\rho_{15}(t)\overline{c_2c_3}\right)e^{-4iat}$

$\frac{d}{dt}\rho_9(t) = ac_1\left(4c_2c_3\rho_2(t) + 4c_2\rho_{14}(t)\overline{c_2} + 4c_3^2\rho_3(t) + 4c_3\rho_{15}(t)\overline{c_2} - 2.0\left(c_2\overline{c_2} + c_3\overline{c_3}\right)\rho_9(t)e^{4iat}\right)e^{-4iat}$

$\frac{d}{dt}\rho_{10}(t) = 2c_0\rho_{10}(t)$

$\frac{d}{dt}\rho_{11}(t) = 2c_0\rho_{11}(t)$

$\frac{d}{dt}\rho_{12}(t) = ac_1\left(-4c_3\rho_2(t)\overline{c_3} + 4c_3\rho_3(t)\overline{c_2} - 2.0\left(c_2\overline{c_2} + c_3\overline{c_3}\right)\rho_{12}(t)e^{4iat} - 4\rho_{14}(t)\overline{c_2c_3} + 4\rho_{15}(t)\overline{c_2}^2\right)e^{-4iat}$

$\frac{d}{dt}\rho_{13}(t) = 4ac_1\left(c_2^2\rho_{10}(t) + c_2c_3\rho_{11}(t) - c_2c_3\rho_6(t) - c_3^2\rho_7(t)\right)$

$\frac{d}{dt}\rho_{14}(t) = c_0\rho_{14}(t)$

$\frac{d}{dt}\rho_{15}(t) = c_0\rho_{15}(t)$

$\frac{d}{dt}\rho_{16}(t) = 4ac_1\left(-c_2\rho_{10}(t)\overline{c_3} + c_2\rho_{11}(t)\overline{c_2} + c_3\rho_6(t)\overline{c_3} - c_3\rho_7(t)\overline{c_2}\right)$

Let me make a few change of variables, and call the new variables c_k

$c_0 = \frac{2.0a\gamma\lambda^2\left(-b_0^2-b_1^2-b_2^2-b_3^2\right)}{16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4}$

$c_1 = \frac{\gamma\lambda^2}{4a^2\gamma^2+\left(4a^2-w_0^2\right)^2}$

$c_2 = b_0 + ib_1$

$c_3 = b_2 + ib_3$

By substituting these into the differential equation we obtain

$\frac{d}{dt}\rho_1(t) = 4ac_1\left(c_2\rho_{10}(t)\overline{c_3} + c_2\rho_6(t)\overline{c_2} + c_3\rho_{11}(t)\overline{c_3} + c_3\rho_7(t)\overline{c_2}\right)$

$\frac{d}{dt}\rho_2(t) = c_0\rho_2(t)$

$\frac{d}{dt}\rho_3(t) = c_0\rho_3(t)$

$\frac{d}{dt}\rho_4(t) = 4ac_1\left(-\rho_{10}(t)\overline{c_3}^2 + \rho_{11}(t)\overline{c_2c_3} - \rho_6(t)\overline{c_2c_3} + \rho_7(t)\overline{c_2}^2\right)$

$\frac{d}{dt}\rho_5(t) = ac_1\left(4c_2^2\rho_2(t) + 4c_2c_3\rho_3(t) - 4c_2\rho_{14}(t)\overline{c_3} - 4c_3\rho_{15}(t)\overline{c_3} - 2.0\left(c_2\overline{c_2} + c_3\overline{c_3}\right)\rho_5(t)e^{4iat}\right)e^{-4iat}$

$\frac{d}{dt}\rho_6(t) = 2c_0\rho_6(t)$

$\frac{d}{dt}\rho_7(t) = 2c_0\rho_7(t)$

$\frac{d}{dt}\rho_8(t) = ac_1\left(-4c_2\rho_2(t)\overline{c_3} + 4c_2\rho_3(t)\overline{c_2} - 2.0\left(c_2\overline{c_2} + c_3\overline{c_3}\right)\rho_8(t)e^{4iat} + 4\rho_{14}(t)\overline{c_3}^2 - 4\rho_{15}(t)\overline{c_2c_3}\right)e^{-4iat}$

$\frac{d}{dt}\rho_9(t) = ac_1\left(4c_2c_3\rho_2(t) + 4c_2\rho_{14}(t)\overline{c_2} + 4c_3^2\rho_3(t) + 4c_3\rho_{15}(t)\overline{c_2} - 2.0\left(c_2\overline{c_2} + c_3\overline{c_3}\right)\rho_9(t)e^{4iat}\right)e^{-4iat}$

$\frac{d}{dt}\rho_{10}(t) = 2c_0\rho_{10}(t)$

$\frac{d}{dt}\rho_{11}(t) = 2c_0\rho_{11}(t)$
$\frac{d}{dt}\rho_{12}(t) = ac_1\left(-4c_3\rho_2(t)\overline{c_3} + 4c_3\rho_3(t)\overline{c_2} - 2.0\left(c_2\overline{c_2} + c_3\overline{c_3}\right)\rho_{12}(t)e^{4iat} - 4\rho_{14}(t)\overline{c_2}\overline{c_3} + 4\rho_{15}(t)\overline{c_2}^2\right)e^{-4iat}$
$\frac{d}{dt}\rho_{13}(t) = 4ac_1\left(c_2^2\rho_{10}(t) + c_2c_3\rho_{11}(t) - c_2c_3\rho_6(t) - c_3^2\rho_7(t)\right)$
$\frac{d}{dt}\rho_{14}(t) = c_0\rho_{14}(t)$
$\frac{d}{dt}\rho_{15}(t) = c_0\rho_{15}(t)$
$\frac{d}{dt}\rho_{16}(t) = 4ac_1\left(-c_2\rho_{10}(t)\overline{c_3} + c_2\rho_{11}(t)\overline{c_2} + c_3\rho_6(t)\overline{c_3} - c_3\rho_7(t)\overline{c_2}\right)$

With The number of symbols reduced the symbolic computation is feasible. However the default solver with initial conditions yields

$$\rho_5(t) = 1.0C_9e^{-t(2.0ac_1c_2\overline{c_2}+2.0ac_1c_3\overline{c_3})} + \left(4.0C_3ac_1c_2^2 + 4.0C_4ac_1c_2c_3 - 4.0C_7ac_1c_2\overline{c_3} - 4.0C_8ac_1c_3\overline{c_3}\right)\left(\begin{cases}\frac{8.0e^{c_0t}e^{-4}}{16.0ac_1c_2\overline{c_2}} \\ t\end{cases}\right.$$

Unfortunately there's a bug in the sympy analytical solver when substituting the initial value conditions to obtain the constants It's not so bad because it is evidend that the weird term is zero. But I check it with manual substitutions anyway below

> **Warning**
> It's only evident if the solution of the equation is a valid density matrix

The initial state considered is

$$\rho(0) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5i & 0 \\ 0 & -0.5i & 0.5 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The solution to the system of equations is

$\rho_1(t) = 1.0C_1 + 1.0C_2e^{2.0c_0t}$
$\rho_2(t) = C_3e^{c_0t}$
$\rho_3(t) = C_4e^{c_0t}$
$\rho_4(t) = 1.0C_5 + 1.0C_6e^{2.0c_0t}$

$$\rho_5(t) = 1.0C_9e^{-t(2.0ac_1c_2\overline{c_2}+2.0ac_1c_3\overline{c_3})} + \left(4.0C_3ac_1c_2^2 + 4.0C_4ac_1c_2c_3 - 4.0C_7ac_1c_2\overline{c_3} - 4.0C_8ac_1c_3\overline{c_3}\right)\left(\begin{cases}\frac{8.0e^{c_0t}e^{-4}}{16.0ac_1c_2\overline{c_2}} \\ t\end{cases}\right.$$

$$\rho_6(t) = \left(\frac{C_{10}c_0c_3\overline{c_3}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2}\overline{c_3}+2.0c_3^2\overline{c_3}^2\right)} - \frac{C_{11}c_0\overline{c_2}\overline{c_3}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2}\overline{c_3}+2.0c_3^2\overline{c_3}^2\right)} + \frac{C_2c_0c_2\overline{c_2}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2}\overline{c_3}+2.0c_3^2\overline{c_3}^2\right)} - \right.$$

$$\rho_7(t) = -\left(\frac{C_{10}c_0c_2\overline{c_3}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2}\overline{c_3}+2.0c_3^2\overline{c_3}^2\right)} + \frac{C_{11}c_0\overline{c_3}^2}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2}\overline{c_3}+2.0c_3^2\overline{c_3}^2\right)} - \frac{C_2c_0c_2\overline{c_3}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2}\overline{c_3}+2.0c_3^2\overline{c_3}^2\right)} - \right.$$

$$\rho_8(t) = 1.0C_{12}e^{-t(2.0ac_1c_2\overline{c_2}+2.0ac_1c_3\overline{c_3})} - \left(4.0C_3ac_1c_2\overline{c_3} - 4.0C_4ac_1c_2\overline{c_2} - 4.0C_7ac_1\overline{c_3}^2 + 4.0C_8ac_1\overline{c_2}\overline{c_3}\right)\left(\begin{cases}\frac{8.0e^{c_0t}e}{16.0ac_1c_2} \\ t\end{cases}\right.$$

$$\rho_9(t) = 1.0C_{13}e^{-t(2.0ac_1c_2\overline{c_2}+2.0ac_1c_3\overline{c_3})} + \left(4.0C_3ac_1c_2c_3 + 4.0C_4ac_1c_3^2 + 4.0C_7ac_1c_2\overline{c_2} + 4.0C_8ac_1c_3\overline{c_2}\right)\left(\begin{cases}\frac{8.0e^{c_0t}e^{-}}{16.0ac_1c_2\overline{c}} \\ t\end{cases}\right.$$

$$\rho_{10}(t) = -\left(\frac{C_{10}c_0c_3\overline{c_2}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2}\overline{c_3}+2.0c_3^2\overline{c_3}^2\right)} - \frac{C_{11}c_0\overline{c_2}^2}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2}\overline{c_3}+2.0c_3^2\overline{c_3}^2\right)} - \frac{C_2c_0c_3\overline{c_2}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2}\overline{c_3}+2.0c_3^2\overline{c_3}^2\right)} + \right.$$

$$\rho_{11}(t) = \left(\frac{C_{10}c_0c_2\overline{c_2}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2}\overline{c_3}+2.0c_3^2\overline{c_3}^2\right)} + \frac{C_{11}c_0\overline{c_2}\overline{c_3}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2}\overline{c_3}+2.0c_3^2\overline{c_3}^2\right)} + \frac{C_2c_0c_3\overline{c_3}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2}\overline{c_3}+2.0c_3^2\overline{c_3}^2\right)} + \right.$$

$$\rho_{12}(t) = 1.0C_{14}e^{-t(2.0ac_1c_2\overline{c_2}+2.0ac_1c_3\overline{c_3})} - \left(4.0C_3ac_1c_3\overline{c_3} - 4.0C_4ac_1c_3\overline{c_2} + 4.0C_7ac_1\overline{c_2}\overline{c_3} - 4.0C_8ac_1\overline{c_2}^2\right)\left(\begin{cases}\frac{8.0e^{c_0t}}{16.0ac_1} \\ t\end{cases}\right.$$

$\rho_{13}(t) = 1.0C_{11}e^{2.0c_0t} + 1.0C_{15}$
$\rho_{14}(t) = C_7e^{c_0t}$
$\rho_{15}(t) = C_8e^{c_0t}$
$\rho_{16}(t) = 1.0C_{10}e^{2.0c_0t} + 1.0C_{16}$

Then by substituting this into the solution we obtain to find the constants we obtain

$\rho_1(t) = \frac{1.0ac_1(c_2\overline{c_2} - ic_2\overline{c_3} + ic_3\overline{c_2} + c_3\overline{c_3})e^{2.0c_0t}}{c_0} - \frac{1.0ac_1(c_2\overline{c_2} - ic_2\overline{c_3} + ic_3\overline{c_2} + c_3\overline{c_3})}{c_0}$
$\rho_2(t) = 0$
$\rho_3(t) = 0$

$$\rho_4(t) = \frac{1.0iac_1\left(\overline{c_2}^2+\overline{c_3}^2\right)e^{2.0c_0t}}{c_0} - \frac{1.0iac_1\left(\overline{c_2}^2+\overline{c_3}^2\right)}{c_0}$$
$$\rho_5(t) = 0$$
$$\rho_6(t) = \frac{c_0\left(-\frac{iac_1c_2c_3\left(\overline{c_2}^2+\overline{c_3}^2\right)}{c_0}+\frac{ac_1c_2(c_2\overline{c_2}-ic_2\overline{c_3}+ic_3\overline{c_2}+c_3\overline{c_3})\overline{c_2}}{c_0}+\frac{1.0ac_1c_3(c_2\overline{c_2}+ic_2\overline{c_3}-ic_3\overline{c_2}+c_3\overline{c_3})\overline{c_3}}{c_0}-\frac{iac_1\left(-c_2^2-c_3^2\right)\overline{c_2c_3}}{c_0}\right)e^{2.0c_0t}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2c_3}+2.0c_3^2\overline{c_3}^2\right)}$$
$$\rho_7(t) = \frac{c_0\left(\frac{iac_1c_2^2\left(\overline{c_2}^2+\overline{c_3}^2\right)}{c_0}+\frac{ac_1c_2(c_2\overline{c_2}-ic_2\overline{c_3}+ic_3\overline{c_2}+c_3\overline{c_3})\overline{c_3}}{c_0}-\frac{1.0ac_1c_2(c_2\overline{c_2}+ic_2\overline{c_3}-ic_3\overline{c_2}+c_3\overline{c_3})\overline{c_3}}{c_0}-\frac{iac_1\left(-c_2^2-c_3^2\right)\overline{c_3}^2}{c_0}\right)e^{2.0c_0t}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2c_3}+2.0c_3^2\overline{c_3}^2\right)}$$
$$\rho_8(t) = 0$$
$$\rho_9(t) = 0$$
$$\rho_{10}(t) = \frac{c_0\left(-\frac{iac_1c_3^2\left(\overline{c_2}^2+\overline{c_3}^2\right)}{c_0}+\frac{ac_1c_3(c_2\overline{c_2}-ic_2\overline{c_3}+ic_3\overline{c_2}+c_3\overline{c_3})\overline{c_2}}{c_0}-\frac{1.0ac_1c_3(c_2\overline{c_2}+ic_2\overline{c_3}-ic_3\overline{c_2}+c_3\overline{c_3})\overline{c_2}}{c_0}+\frac{iac_1\left(-c_2^2-c_3^2\right)\overline{c_2}^2}{c_0}\right)e^{2.0c_0t}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2c_3}+2.0c_3^2\overline{c_3}^2\right)}$$
$$\rho_{11}(t) = \frac{c_0\left(\frac{iac_1c_2c_3\left(\overline{c_2}^2+\overline{c_3}^2\right)}{c_0}+\frac{1.0ac_1c_2(c_2\overline{c_2}+ic_2\overline{c_3}-ic_3\overline{c_2}+c_3\overline{c_3})\overline{c_2}}{c_0}+\frac{ac_1c_3(c_2\overline{c_2}-ic_2\overline{c_3}+ic_3\overline{c_2}+c_3\overline{c_3})\overline{c_3}}{c_0}+\frac{iac_1\left(-c_2^2-c_3^2\right)\overline{c_2c_3}}{c_0}\right)e^{2.0c_0t}}{ac_1\left(2.0c_2^2\overline{c_2}^2+4.0c_2c_3\overline{c_2c_3}+2.0c_3^2\overline{c_3}^2\right)}$$
$$\rho_{12}(t) = 0$$
$$\rho_{13}(t) = \frac{1.0iac_1\left(-c_2^2-c_3^2\right)e^{2.0c_0t}}{c_0} - \frac{1.0iac_1\left(-c_2^2-c_3^2\right)}{c_0}$$
$$\rho_{14}(t) = 0$$
$$\rho_{15}(t) = 0$$
$$\rho_{16}(t) = \frac{1.0ac_1(c_2\overline{c_2}+ic_2\overline{c_3}-ic_3\overline{c_2}+c_3\overline{c_3})e^{2.0c_0t}}{c_0} - \frac{1.0ac_1(c_2\overline{c_2}+ic_2\overline{c_3}-ic_3\overline{c_2}+c_3\overline{c_3})}{c_0}$$

By reverting the change of variables. The analytical answer of the Bloch redfield equation is given by

$$\rho_1(t) = \frac{0.5\left(e^{\frac{4.0a\gamma\lambda^2t\left(b_0^2+b_1^2+b_2^2+b_3^2\right)}{16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4}}-1\right)\left(16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4\right)((b_0-ib_1)(b_0+ib_1)+i(b_0-ib_1)(b_2+ib_3)-i(b_0+ib_1)(b_2-\cdots}{\left(4a^2\gamma^2+\left(4a^2-w_0^2\right)^2\right)\left(b_0^2+b_1^2+b_2^2+b_3^2\right)\cdots}$$
$$\rho_2(t) = 0$$
$$\rho_3(t) = 0$$
$$\rho_4(t) = \frac{0.5i\left((b_0-ib_1)^2+(b_2-ib_3)^2\right)\left(e^{\frac{4.0a\gamma\lambda^2t\left(b_0^2+b_1^2+b_2^2+b_3^2\right)}{16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4}}-1\right)\left(16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4\right)e^{-\frac{4.0a\gamma\lambda^2t\left(b_0^2+b_1^2+b_2^2+b_3^2\cdots\right.}{16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1\cdots}}}{\left(4a^2\gamma^2+\left(4a^2-w_0^2\right)^2\right)\left(b_0^2+b_1^2+b_2^2+b_3^2\right)\cdots}$$
$$\rho_5(t) = 0$$
$$\rho_6(t) = 0.5e^{-\frac{4.0a\gamma\lambda^2t\left(b_0^2+b_1^2+b_2^2+b_3^2\right)}{16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4}}$$
$$\rho_7(t) = 0.5ie^{-\frac{4.0a\gamma\lambda^2t\left(b_0^2+b_1^2+b_2^2+b_3^2\right)}{16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4}}$$
$$\rho_8(t) = 0$$
$$\rho_9(t) = 0$$
$$\rho_{10}(t) = -0.5ie^{-\frac{4.0a\gamma\lambda^2t\left(b_0^2+b_1^2+b_2^2+b_3^2\right)}{16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4}}$$
$$\rho_{11}(t) = 0.5e^{-\frac{4.0a\gamma\lambda^2t\left(b_0^2+b_1^2+b_2^2+b_3^2\right)}{16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4}}$$
$$\rho_{12}(t) = 0$$
$$\rho_{13}(t) = \frac{0.5i\left(1-e^{\frac{4.0a\gamma\lambda^2t\left(b_0^2+b_1^2+b_2^2+b_3^2\right)}{16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4}}\right)\left((b_0+ib_1)^2+(b_2+ib_3)^2\right)\left(16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4\right)e^{-\frac{4.0a\gamma\lambda^2t\left(b_0^2+b_1^2+b_2^2+b\cdots\right.}{16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+\cdots}}}{\left(4a^2\gamma^2+\left(4a^2-w_0^2\right)^2\right)\left(b_0^2+b_1^2+b_2^2+b_3^2\right)\cdots}$$
$$\rho_{14}(t) = 0$$
$$\rho_{15}(t) = 0$$

$$\rho_{16}(t) = \frac{0.5\left(e^{\frac{4.0a\gamma\lambda^2 t\left(b_0^2+b_1^2+b_2^2+b_3^2\right)}{16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4}}-1\right)\left(16.0a^4+4.0a^2\gamma^2-8.0a^2w_0^2+1.0w_0^4\right)((b_0-ib_1)(b_0+ib_1)-i(b_0-ib_1)(b_2+ib_3)+i(b_0+ib_1)(b_2}{\left(4a^2\gamma^2+\left(4a^2-w_0^2\right)^2\right)\left(b_0^2+b_1^2+b_2^2+b_3^2\right)}$$

We can then substitute the numerical values for example for the case we explored above

$$\rho(t) = \begin{bmatrix} 0.55247613848278\left(e^{0.94679965816826t}-1.0\right)e^{-0.94679965816826t} & 0 \\ 0 & 0.5e^{-0.946799} \\ 0 & -0.5ie^{-0.9467} \\ (-0.486572940196368-0.102435485836685i)\left(1.0-e^{0.94679965816826t}\right)e^{-0.94679965816826t} & 0 \end{bmatrix}$$

Then we may evaluate for long times

$$\rho(50) = \begin{bmatrix} 0.55247613848278 & 0 & 0 & 0.486572940196368-0.102435485836685i \\ 0 & 0.0 & 0 & 0 \\ 0 & 0 & 0.0 & 0 \\ 0.486572940196368+0.102435485836685i & 0 & 0 & 0.44752386151722 \end{bmatrix}$$

I believe I was careful enough to use the same convention used in the other equations (Pseudomodes, Cumulant and redfield) but currently reviewing the derivations to make sure there's no inconsistencies. The derivations in question are in https://master–gsuarezthesis.netlify.app/redfield . I do think it is now safe to assume that BR/Redfield breaksdown and that it is not a bug in the code, so maybe we can write a paper on redfield breaking down, cumulant/global being good once we figure out why it happens. Though it seems to be about the coupling and not the degeneracies

About Pictures
Technically the above matrix is not correct as it is in the interaction picture and not the Schrodinger picture. In this case it does not make a difference, however, let us do the rotation

```
U=exp(I*H*t)
```

$$U = \begin{bmatrix} e^{-iat} & 0 & 0 & 0 \\ 0 & e^{iat} & 0 & 0 \\ 0 & 0 & e^{iat} & 0 \\ 0 & 0 & 0 & e^{-iat} \end{bmatrix}$$

$$U\rho U^{\dagger} = \begin{bmatrix} \rho_1 & \rho_2 e^{-2iat} & \rho_3 e^{-2iat} & \rho_4 \\ e^{2iat}\overline{\rho_2} & \rho_6 & \overline{\rho_{10}} & \rho_8 e^{2iat} \\ e^{2iat}\overline{\rho_3} & \rho_{10} & \rho_{11} & \rho_{12} e^{2iat} \\ \overline{\rho_4} & e^{-2iat}\overline{\rho_8} & e^{-2iat}\overline{\rho_{12}} & -\rho_1-\rho_{11}-\rho_6+1 \end{bmatrix}$$

```
rhoss=U*roundMatrix(ans.subs(num_values).subs(t,150).evalf(),18)*Dagger(U)
```

$$\rho(50) = \begin{bmatrix} 0.55247613848278 & 0 & 0 & 0.486572940196368-0.102435485836685i \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0.486572940196368+0.102435485836685i & 0 & 0 & 0.44752386151722 \end{bmatrix}$$

0.1.1 RC picture of the Hamiltonian

For the RC I simply follow one of your papers https://arxiv.org/pdf/1511.05181

So If I didn't misunderstand it then

$$\Omega = w_0 \tag{8}$$

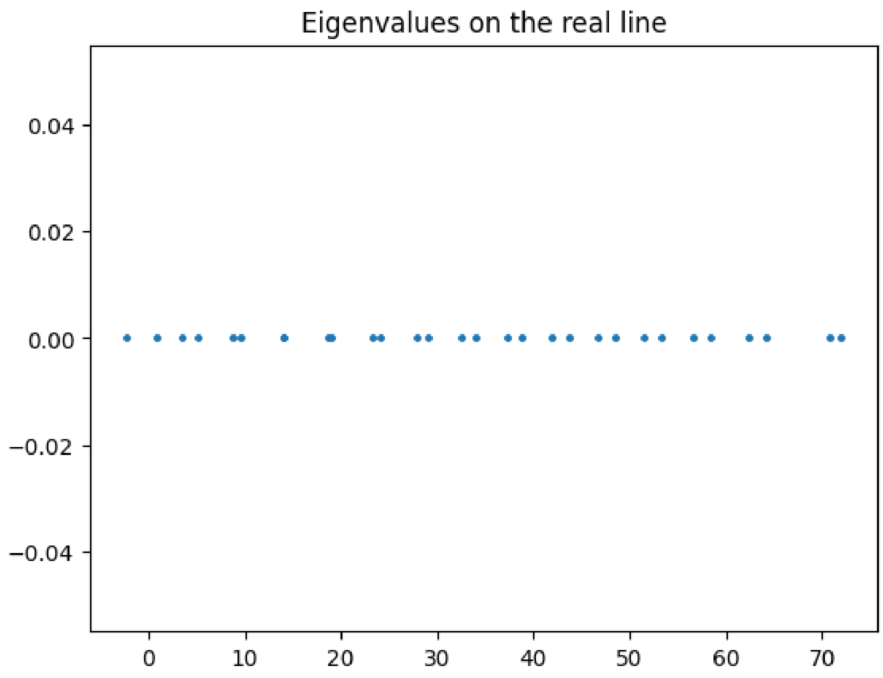
$$\lambda_{rc} = \sqrt{\frac{\pi}{2w_0}}\lambda \tag{9}$$

Not actually sure if π should be there, but does not seem to be relevant for the question we are asking

Not sure how many levels to take here, but let us guess 15 is enough
Then I construct the RC Hamiltonian

```
NHRC=HRC.subs(num_values).evalf()
```

```
plt.scatter(np.real(eigenvalues),np.round(np.imag(eigenvalues),10),s=5)
plt.title("Eigenvalues on the real line")
plt.show()
```



Propably I shopuld have done the partial trace so

```
qHRC.ptrace(0)
```

```
Quantum object: dims=[[4], [4]], shape=(4, 4), type='oper', isherm=True
Qobj data =
[[471.25867623   0.           0.           0.        ]
 [  0.         536.71867623   0.           0.        ]
 [  0.           0.         536.71867623   0.        ]
 [  0.           0.           0.         471.25867623]]
```

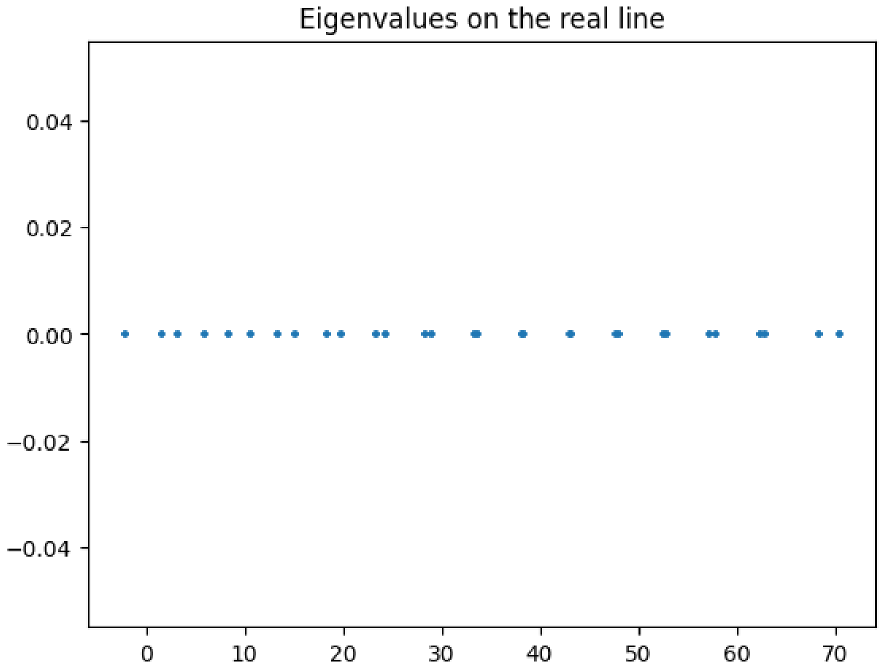
Still degenerate
Let us try the other coupling which does not break bloch redfield

```
qHRC.ptrace(0)
```

```
Quantum object: dims=[[4], [4]], shape=(4, 4), type='oper', isherm=True
Qobj data =
[[471.25867623   0.           0.           0.        ]
 [  0.         536.71867623   0.           0.        ]
 [  0.           0.         536.71867623   0.        ]
 [  0.           0.           0.         471.25867623]]
```

```
eigenvalues, eigenvectors = np.linalg.eig(ans)
```

```
plt.scatter(np.real(eigenvalues), np.round(np.imag(eigenvalues), 10), s=5)
plt.title("Eigenvalues on the real line")
plt.show()
```



There doesn't seem to be much change in the Hamiltonian if any

0.2 Hamiltonian Simulation

0.2.1 When the Hamiltonian is Physical

When the Hamiltonian is physical we can take the same steps as people who simulate the Lindblad master equation. I followed the Lin Lin paper. Here's a bit of the paper but instead of using SDE schemes, I do it on the ensemble. To obtain the Krauss operators (I also only do it to first order, because it is simplest and should work when $dt \to 0$ but probably one should consider higher orders).

0.2.2 Krauss Operator from a Lindbladian

> **Warning**
> This might be a mistake from the very beginning as Pseudomodes is not a CPTP map.However numerically, I've never seen any issue with positivity when enough levels are considered in the modes. So from here I'm assumming it will be CPTP

The logic here is to start from the master equation

$$\dot{\rho(t)} = \mathcal{L}(\rho(t)) \tag{10}$$

From the definition of derivative this means

$$\lim_{dt\to 0} \frac{\rho(t+dt)-\rho(t)}{dt} = \mathcal{L}(\rho(t)) \tag{11}$$

For now let us forget about the limit, but work our quantities approximately and to order $\mathcal{O}(dt^2)$ so that

$$\rho(t+dt) \approx \rho(t) + \mathcal{L}(\rho(t))dt + \mathcal{O}(dt^2) \tag{12}$$

Since the map is CPTP then it must have a sum operator representation (Krauss representation) so

$$\rho(t+dt) = \sum_k M_k \rho(t) M_k^\dagger \tag{13}$$

What's left now is to find what the krauss operators should be. This is known and can be seen for example in Lidar's lecture notes. Here I do some extra algebra to illustrate how to generalize to higher order schemes

Since this is lowest order then, we propose the Krauss operators

$$M_0 = \infty + Adt \tag{14}$$

$$M_k = \sqrt{dt} B_k \tag{15}$$

Then we find that

> **Warning**
> I did these calculations by hand, but I am really lazy when it comes to latexing so I decided to use sympy for intermediate steps, if there's any inconsistency I can just latex those steps

$\rho(dt+t) = \sum_{k=1}^{N} dt B_k \rho(t) B_k^\dagger + \rho(t) + dt\rho(t)A^\dagger + dtA\rho(t) + O\left(dt^2\right)$

Notice the series of Krauss operators we used have identical contributions (in their form, and could be represented as a sum). Notice on the other hand we have

$$\rho(t+dt) \approx \rho(t) + \mathcal{L}(\rho(t))dt + \mathcal{O}(dt^2) \tag{16}$$

By replacing the lindbladian one obtains

$$\rho(t+dt) \approx \rho(t) + \left(-i[H,\rho(t)] + \sum_k L_k \rho(t) L_k^\dagger - \frac{\{L_k^\dagger L_k, \rho(t)\}}{2}\right) dt \tag{17}$$

where the L_k are the jump operators. Then by comparison we can find the required Krauss operators, First let us note That to generate a commutator A must be an Anti-Hermitian matrix, and to generate an anticommutator A must be Hermitian. If we choose A to be a sum of a Hermitian and Anti-hermitian matrix then

$A = -iH + K$

$\rho(dt+t) = \sum_{k=1}^{N} dt B_k \rho(t) B_k^\dagger + \rho(t) + dt\rho(t)K + idt\rho(t)H + dtK\rho(t) - idtH\rho(t) + O\left(dt^2\right)$

Which we can simplify to

$\rho(dt+t) = \sum_{k=1}^{N} dtL\rho(t)L^\dagger + \rho(t) - \frac{dt\left(\sum_{k=1}^{N} L_k^\dagger L_k\right)\rho(t)}{2} - \frac{dt\rho(t)\sum_{k=1}^{N} L_k^\dagger L_k}{2} + idt\rho(t)H - idtH\rho(t) + O\left(dt^2\right)$

At this point notice that if we select the B_k to be the jump operators and K to be the corresponding anticommutator term

$\rho(dt+t) = \sum_{k=1}^{N} dtL\rho(t)L^\dagger + \rho(t) - \frac{dt\left(\sum_{k=1}^{N} L_k^\dagger L_k\right)\rho(t)}{2} - \frac{dt\rho(t)\sum_{k=1}^{N} L_k^\dagger L_k}{2} + idt\rho(t)H - idtH\rho(t) + O\left(dt^2\right)$

We have obtained the First order scheme to obtain the Linblad master equation Krauss operators. To obtain Higher order schemes One can notice that the solution to the master equation is

$$\rho(t) = e^{\mathcal{L}t}\rho(0) \tag{18}$$

and

$$\rho(t+dt) = e^{\mathcal{L}(t+dt)}\rho(0) = e^{\mathcal{L}dt}\rho(t) \tag{19}$$

And then expand the series of the exponential. Similarly one should increase the order of the krauss operator guess by one and find the appropiate operators

0.2.3 Notice We could have not guess K and use the Completeness relation of krauss operators

In this case it was not needed but it might be useful to find relations in higher order schemes and to check the krauss operators are ok. So next we find K this way

The completeness relation indicates

$$\sum_k M_k^\dagger M_k = 1 \tag{20}$$

Since in our schemes we are numerically approximating to $\mathcal{O}(dt^2)$ then

$$\sum_k M_k^\dagger M_k = 1 + \mathcal{O}(dt^2) \tag{21}$$

$1 = 1 + dt\sum_{k=1}^{N} L_k^\dagger L_k + 2dtK + O\left(dt^2\right)$

Which we can solve to find

$K = -\frac{\sum_{k=1}^{N} L_k^\dagger L_k}{2} + O\left(dt\right)$

Now that we have found the Krauss operators one may simply ask if one can follow the same scheme to obtain the Krauss operators of a pseudomode equation. Since I have not seen positivity issues I do think it's possible. but the naive approach to it yields and inconsistency

0.2.4 Same derivation with a non-Hermitian Hamiltonian

Any non-Hermitian matrix can be split into the sum of a Hermitian and Anti-Hermitian Matrix such that I can write the unphysical Hamiltonian H as

$$H = H_0 + iH_u \tag{22}$$

Then the Lindblad equation turns into

$$\rho(t+dt) \approx \rho(t) + \left(-i[H_0, \rho(t)] + [H_u, \rho(t)] + \sum_k L_k \rho(t) L_k^\dagger - \frac{\{L_k^\dagger L_k, \rho(t)\}}{2} \right) dt \tag{23}$$

Following the same strategy as before only A changes (the part that generated the commutator), so the change is only on M_0 We neglect the part that contains K as that one does not change

$\rho(t) + idt\rho(t)H^\dagger - idtH\rho(t) + O\left(dt^2\right)$

Substitute

$$H = H_0 + iH_u \tag{24}$$

$\rho(t) + dt\left(H_u\rho(t) + \rho(t)H_u - i\left[H_0, \rho(t)\right]\right) + O\left(dt^2\right)$

By comparison we would need

$$\{H_u, \rho(t)\} = [H_u, \rho(t)] \tag{25}$$

Which cannot be satisfied

Even though this calculation was a failure. Perhaps one would need to use higher orders, or reorder the terms in another Fashion. I do believe it would be easier if I don't use the general formulation but the jump operators as a and $a^\dagger$ and the unphysical part of the Hamiltonian to have the form $\sum_k \omega_k a_k^\dagger a_k$ where ω_k is complex.

While I look into it. Asume the Hamiltonian is Physical and then extrapolation is done. One has several schemes to simulate Krauss operators in a quantum circuit simulator. Let us go with the scheme in the Lin Lin paper

Even though in their case is not so bad, Here I illustrate why I don't like this Hamiltonian approach in the first order. Perhaps it is better to use the other 2nd Lin Lin paper though I think the number of ancillas needed will be bigger. I also need to try the Hush et al. [2015]

0.2.5 Hamiltonian Simulation From Krauss Operators

In this section all traces are partial traces with respect to the ancilla

The paper suggests using the Stinespring representation of the Krauss Operators Namely. Finding an ancilla and a matrix μ such that

$$Tr(\mu) = \sum_k M_k \rho(t) M_k^\dagger \tag{26}$$

To achieve this we can use the matrix μ

$$\mu = \begin{pmatrix} M_0\rho(t)M_0^\dagger & M_0\rho(t)M_1^\dagger & \dots & M_0\rho(t)M_k^\dagger \\ M_1\rho(t)M_0^\dagger & M_1\rho(t)M_1^\dagger & \dots & M_1\rho(t)M_k^\dagger \\ \vdots & \vdots & \ddots & \vdots \\ M_k\rho(t)M_0^\dagger & \dots & \dots & M_k\rho(t)M_k^\dagger \end{pmatrix} \tag{27}$$

Which can be easily constructed by requiring one ancilla qubit for each jump operator. We consider all the ancillas to be on the ground state such that the state of the ancillas is

$00 = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix}$

Then one can obtain μ by

$$\mu = A00 \otimes \rho A^\dagger \tag{28}$$

Where

$$A = \begin{pmatrix} M_0 & M_1^\dagger & \dots & M_k^\dagger \\ M_1 & 0 & \dots & 0 \\ \vdots & 0 & \dots & 0 \\ M_k & 0 & \dots & 0 \end{pmatrix} \tag{29}$$

Then one can express the operator sum representation as

$$Tr(A00 \otimes \rho A^\dagger) = \sum_k M_k \rho(t) M_k^\dagger \tag{30}$$

To have a Hamiltonian simulation of the Krauss representation we want to find a unitary such that

$$Tr(U00 \otimes \rho U^\dagger) = Tr(A00 \otimes \rho A^\dagger) = \sum_k M_k \rho(t) M_k^\dagger \tag{31}$$

Or at least to order $\mathcal{O}(dt^2)$. One of the insights of the paper is to write U as

$$U = e^{-i\sqrt{dt}\bar{H}} \tag{32}$$

Where

$$\bar{H} = \begin{pmatrix} H_0 & H_1^\dagger & \dots & H_k^\dagger \\ H_1 & 0 & \dots & 0 \\ \vdots & 0 & \dots & 0 \\ H_k & 0 & \dots & 0 \end{pmatrix} \tag{33}$$

with Hermitian H_0. Then One can obtain $\bar{H}$ from taylor expanding the exponential and matching the same order terms. For simplicity here I do it for 4 jump operators

While at first order in the exponential we have

$$exp(x) \approx 1 + x \tag{34}$$

$$U = \begin{bmatrix} -i\sqrt{dt}H_0 + 1 & -i\sqrt{dt}H_1^\dagger & -i\sqrt{dt}H_2^\dagger & -i\sqrt{dt}H_3^\dagger & -i\sqrt{dt}H_4^\dagger \\ -i\sqrt{dt}H_1 & 1 & 0 & 0 & 0 \\ -i\sqrt{dt}H_2 & 0 & 1 & 0 & 0 \\ -i\sqrt{dt}H_3 & 0 & 0 & 1 & 0 \\ -i\sqrt{dt}H_4 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Again the goal is to have this emulate the krauss operators, which emulate the master equation

$\rho(dt + t) = \sum_{k=1}^N dtL\rho(t)L^\dagger + \rho(t) - \frac{dt\left(\sum_{k=1}^N L_k^\dagger L_k\right)\rho(t)}{2} - \frac{dt\rho(t)\sum_{k=1}^N L_k^\dagger L_k}{2} + idt\rho(t)H - idtH\rho(t) + O\left(dt^2\right)$

By using

$U\rho U^\dagger = \sqrt{dt}\left(i\rho(t)H_0 - iH_0\rho(t)\right) + dt\left(H_0\rho(t)H_0 + H_1\rho(t)H_1^\dagger + H_2\rho(t)H_2^\dagger + H_3\rho(t)H_3^\dagger + H_4\rho(t)H_4^\dagger\right) + \rho(t)$

We further simiplify it to be

$U\rho U^\dagger = -i\sqrt{dt}\left[H_0, \rho(t)\right] + dt\left(H_0\rho(t)H_0 + \sum_{k=1}^4 H_k\rho(t)H_k\right) + \rho(t)$

To approximate the master equation notice that we can have

$H_0 = \sqrt{(}dt)H$

$H_k = L_k$

Which is the first order in the Lin Lin paper and results in

$U\rho U^\dagger = dt\left(dtH\rho(t)H + \sum_{k=1}^4 L_k\rho(t)L_k\right) - idt\left[H, \rho(t)\right] + \rho(t)$

Then neglecting higher order terms one has
$U\rho U^{\dagger} = \rho(t) + dt \sum_{k=1}^{4} L_k \rho(t) L_k - i dt \left[H, \rho(t)\right] + O\left(dt^2\right)$
But this neglects the anticommutator bit This is what I don't like but it can be fixed by higher orders does not seem to affect accuracy too much

TO DO!

- Use higher order Schemes, and try to have a non-Hermitian Hamiltonian
- Test pseudomodes being CPTP by calculating $\tau := (\mathcal{L} \otimes \mathrm{id}_d)(|\Omega\rangle\langle\Omega|)$ where Ω is the maximally entangled state (if CPTP $\tau \geq 0$)
- Check other simulation schemes like the other lin lin paper or Clover's paper
- Actually do the Hamiltonian simulation, The ancillas need to be reset every timestep but this suceeds with probabulity one in theory

Bibliography

M. R. Hush, I. Lesanovsky, and J. P. Garrahan. Generic map from non-Lindblad to Lindblad master equations. *Physical Review A*, 91(3), 3 2015. ISSN 1094-1622. doi: 10.1103/physreva.91.032113. URL `http://dx.doi.org/10.1103/PhysRevA.91.032113`.