

PES UNIVERSITY
Karnataka, Bangalore-560080



Course: Non-linear Optics and Quantum technology

Course code: UE22EC343BB1

AHP Report

“Optical quantum computing using cubic phase gates”

REPORT BY:

Name: Nithin M SRN: PES1UG22EC184

Name: Purab P Bhat SRN: PES1UG22EC220

Under guidance of:

Dr Kaustav Bhowmick

ECE Dept



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING**

1.Abstract: For universal quantum computation non-clifford gates are essential, enabling operations beyond the Clifford group. In optical quantum computing, the cubic phase gate is the canonical non-Gaussian non-Clifford gate that introduces the necessary nonlinearity for universal continuous-variable (CV) processing. This report explores optical non-Clifford gates, focusing on the design and implementation of a cubic phase gate using three-wave mixing in nonlinear optical materials. A hybrid decomposition approach combining beam splitters and single-mode cubic phase gates was analyzed, and design considerations using experimentally feasible parameters are discussed.

2.Introduction:

A. Non – Clifford Gate

Universal quantum computation with photonic systems offers compelling advantages, including scalability and potential room-temperature operation. However, achieving full computational universality in photonics requires operations beyond the Gaussian (Clifford) regime. In the continuous-variable (CV) framework, Clifford operations—such as beam splitters, phase shifters, and squeezers—are easily implemented but remain classically simulatable and thus insufficient for universal quantum computing.

To overcome this limitation, the introduction of non-Gaussian, non-Clifford operations is essential. The **cubic phase gate**, mathematically expressed as

$$V(\gamma) = e^{i\gamma\hat{x}^3}$$

stands as the canonical single-mode non-Clifford gate. This gate induces nonlinear transformations on the optical field's quadratures, enabling operations that cannot be efficiently simulated on classical computers. Specifically, under the action of the cubic phase gate, the momentum operator transforms as

$$V(\gamma)^\dagger \hat{p} V(\gamma) = \hat{p} + 3\gamma\hat{x}^2$$

introducing critical non-Gaussian features into the quantum state.

Recent studies, particularly by Budinger et al., highlight the pivotal role of the cubic phase gate in achieving universal CV quantum computation. They demonstrate that any arbitrary CV quantum gate, including complex multimode operations, can be decomposed into a finite sequence of linear optical elements and identical cubic phase gates. Thus, while linear optics alone generates only Clifford operations, the integration of a non-Gaussian resource like the cubic phase gate completes the universal set.

Beyond enabling universal computation, cubic phase gates combined with linear optics facilitate the construction of important entangling operations, such as two-mode controlled-phase gates and three-mode Rabi-type interactions. These are foundational for creating highly entangled optical states, including Gottesman–Kitaev–Preskill (GKP) states, which are essential for fault-tolerant quantum computing.

Nevertheless, the practical realization of cubic phase gates presents formidable challenges. Natural optical nonlinearities, such as three-wave mixing ($\chi^{(2)}$) and Kerr effects ($\chi^{(3)}$) can theoretically induce the required cubic interactions but are typically too weak or lossy for scalable, fault-tolerant applications. Although nonlinear feedforward techniques—using squeezed-light ancillae and adaptive measurements—have achieved partial experimental demonstrations, integrating strong, low-loss non-Gaussian operations remains an open problem. Active research efforts are directed toward exploiting engineered χ^2 materials and designing photonic circuits that can directly implement cubic nonlinearities through three-wave mixing processes.

In this context, this report aims to study the theoretical foundation of optical non-Clifford gates, focusing on the cubic phase gate, and to explore a feasible design for its realization via three-wave mixing in nonlinear photonic platforms.

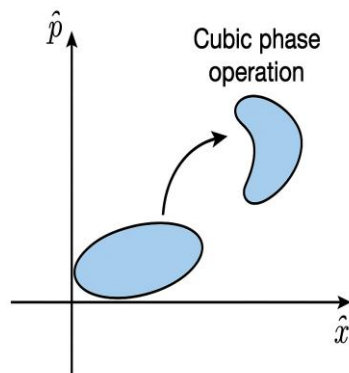


Fig1: Optical non-clifford gate

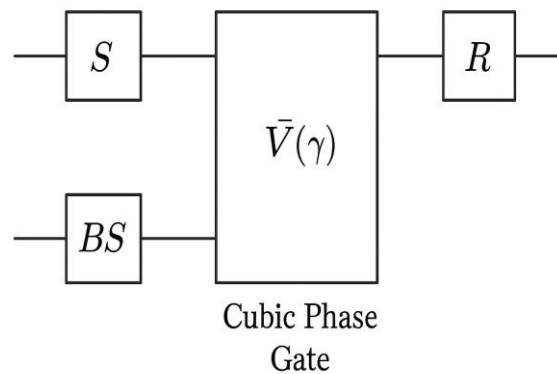


Fig2: Cubic phase gate deforming Squeezed state.

Properties of Optical Non-Clifford Gates

- Non-Gaussian: Introduce nonlinearity beyond simple squeezing, displacement, and rotations.
- Essential for Universality: Required to go beyond Clifford operations.
- Hard to Implement: Need strong optical nonlinearities, often approximated via ancillary photons, squeezed states, or measurement-induced methods.

Optical non-Clifford gates like the cubic phase gate are crucial for scalable, universal, and fault-tolerant quantum computing. Although challenging to implement, new advances in nonlinear optics and quantum optics promise increasingly practical methods to realize these operations.

Working of Optical Non-Clifford Gates

Optical non-Clifford Gates are quantum gates that cannot be built from Clifford gates (Pauli, Hadamard, Phase) and are essential for universal quantum computation. Unlike Clifford gates, which can be efficiently simulated classically, non-Clifford gates enable operations beyond simple rotations, often involving complex entanglement.

In optical quantum computing, these gates are typically realized through nonlinear optical processes like three-wave mixing, using quantum interference and superposition in nonlinear fibers or crystals. For example, the T-gate ($\pi/4$ rotation) can be implemented via nonlinear phase shifts that manipulate photon states in ways Clifford operations cannot.

In short, optical non-Clifford gates use nonlinear optics to achieve the complex quantum transformations needed for universal computation.

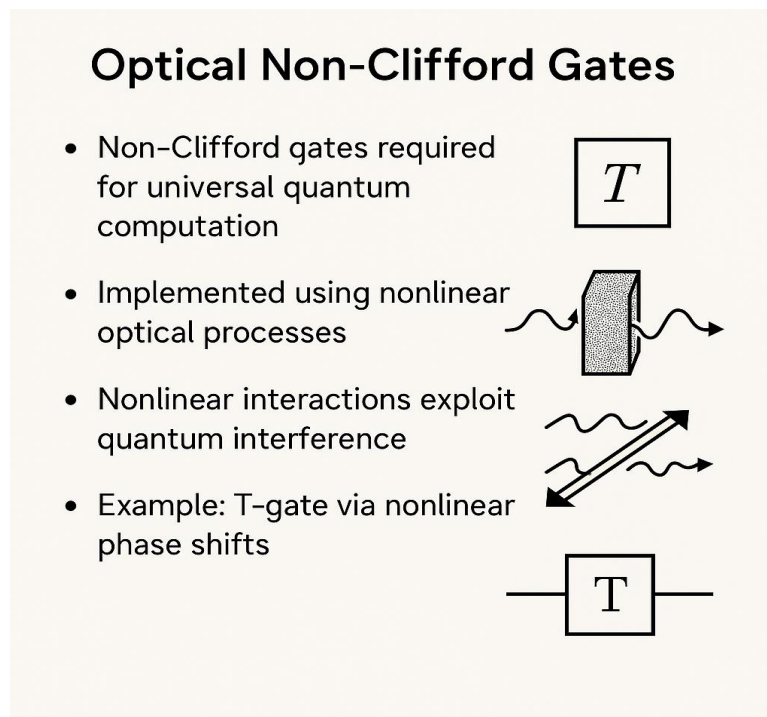


Fig3: Overview of optical implementation of non-Clifford gate

2. Theoretical model:

- Material: Periodically poled lithium niobate (PPLN) for efficient second-order nonlinearity.
- Pump: Continuous-wave laser at telecom or visible wavelength.
- Ancilla: A weak cubic phase state prepared using photon subtraction from squeezed states.

3. Implementation Scheme:

- Generate a squeezed vacuum state.
- Apply displacement operations.
- Use three-wave mixing to approximate a nonlinear Hamiltonian resembling x^3 .
- Employ homodyne detection and classical feedforward for error mitigation.

(B). Phase and Cubic Phase Gates

a. Phase rotation gates.

In quantum optics, a phase gate rotates the state in phase space:

$$\hat{R}(\theta) = e^{i\theta\hat{n}}$$

where $\hat{n} = \hat{a}^\dagger\hat{a}$ is the number operator.

- A rotation gate simply multiplies each Fock state by a phase:

$$\hat{R}(\theta)|n\rangle = e^{in\theta}|n\rangle$$

These are Gaussian and Clifford gates — easy to implement with beam splitters, mirrors, simple optics.

b. Cubic Phase Gate

- A cubic phase gate is non-Gaussian and non-Clifford, essential for universal continuous-variable (CV) quantum computation:

$$\hat{V}(\gamma) = e^{i\gamma\hat{x}^3}$$

where x is the position quadrature:

$$\hat{x} = \frac{1}{\sqrt{2}}(\hat{a} + \hat{a}^\dagger)$$

- Unlike simple phase gates, the cubic gate warps the quantum state's Wigner function nonlinearly.
- Challenge: Cubic transformations need nonlinear media or nonlinear circuits — not just passive optics.

c. Three-Wave Mixing (χ^2 Nonlinearity)

- Three-wave mixing happens in materials with second-order susceptibility χ^2 .
- If you have two strong optical (or microwave) pumps at frequencies ω_1 and ω_2 , you can create or destroy a third signal at ω_3 such that:

$$\omega_3 = \omega_1 \pm \omega_2$$

- Example devices:
 - Nonlinear optical crystals (e.g., Lithium Niobate, KTP)
 - Josephson-based circuits (e.g., SNAILs, JRMJs) for microwave frequencies
- Three-wave mixing can create effective cubic nonlinearities needed for cubic gates when engineered carefully.

d. Materials and Properties

- Material: Lithium Niobate (LiNbO_3)
 - $\chi^2 \approx 20\text{--}30$ pm/V (strong)
 - Wide transparency window (visible to telecom wavelengths)
 - Available as thin films (LNOI: Lithium Niobate on Insulator)
- Peripheral Devices:
 - Waveguides: Ridge waveguides in LNOI, cross-section: 500×1000 nm.
 - Resonators: Microring or microdisk resonators for field enhancement.
 - Pumps: Two lasers phase-locked at ω_1 and ω_2 .
- Phase Matching: Need periodic poling (PPLN) to quasi-phase-match frequencies.

e. Implementation

Basic Hamiltonian (Superconducting version)

- Effective Hamiltonian for the cubic phase interaction:

$$H = \hbar\omega_r \hat{a}^\dagger \hat{a} + \hbar g (\hat{a} + \hat{a}^\dagger)^3$$

- ω_r = resonator frequency (e.g., 6 GHz)
- g = cubic coupling strength (maximize)
- Control: By adjusting flux bias, pump tones, and coupling rates, you can tune the strength g .

Target

- We want to design a system such that applying it for a time t realizes:

$$U_{\text{cubic}} = e^{i\gamma \hat{x}^3} \quad \text{with} \quad \gamma = gt$$

- Tune g as large as possible (strong nonlinearity)
- Choose t appropriately (e.g., nanoseconds)
- Minimize loss during operation

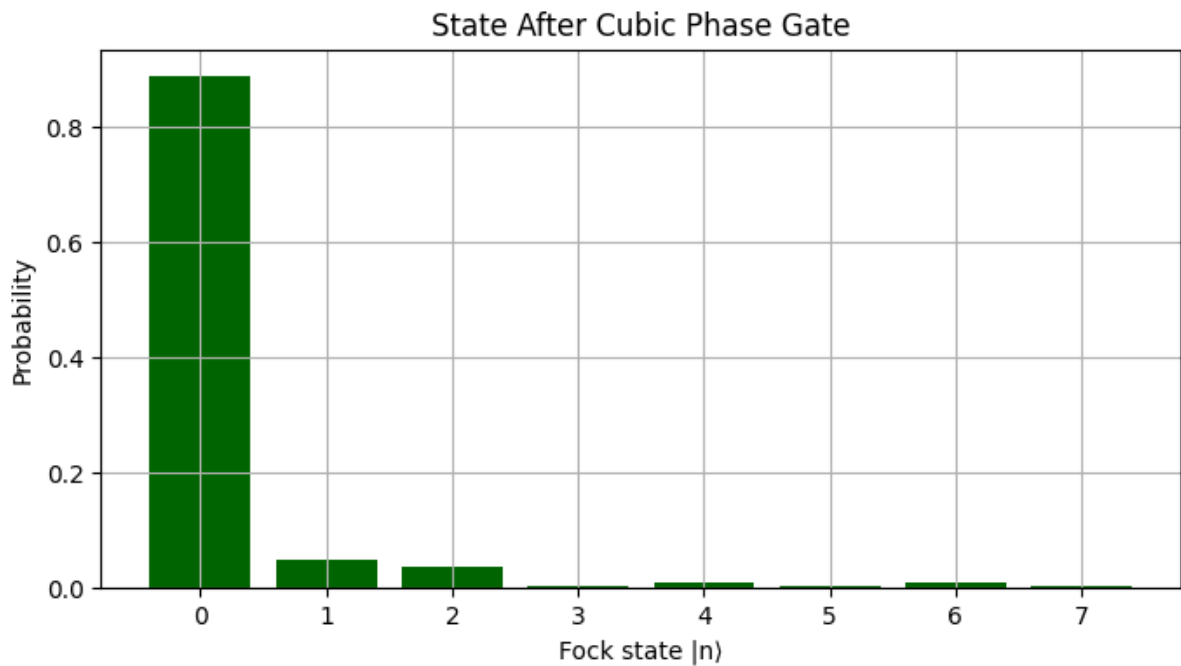


Fig 4: Qiskit-Only Cubic Phase Gate Visualization.

3. Result

Design of low-loss, strongly nonlinear, phase-matched setup that can approximate a cubic phase gate sufficiently for CV quantum computing or magic state generation(shown in fig 5).

(i)Initial State

- You start with some quantum state $|\psi\rangle$

(ii)First Hadamard (H)

- Hadamard gate transforms:

$$H|0\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \quad H|1\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

- It puts the qubit into a superposition.

(iii)Phase Gate $P(\pi/4)$

- The Phase gate applies a phase shift of θ to $|1\rangle$ only:

$$P(\theta)|0\rangle = |0\rangle, \quad P(\theta)|1\rangle = e^{i\theta}|1\rangle$$

- For $\theta=\pi/4$, this gives a small phase rotation only to the $|1\rangle$ component.

(iv)Second Hadamard (H)

- Another Hadamard mixes the amplitudes again.

(v)Net Effect

- This whole sequence acts like a rotation around the X-axis of the Bloch sphere.
- In fact, it's equivalent to an $RX(\pi/4)$ rotation up to a global phase.
($RX(\theta)$ is the rotation around the X-axis by angle θ .)

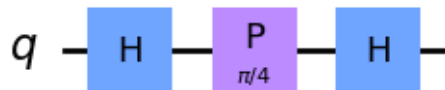


Fig 5: Phase Gate $P(\pi/4)$.

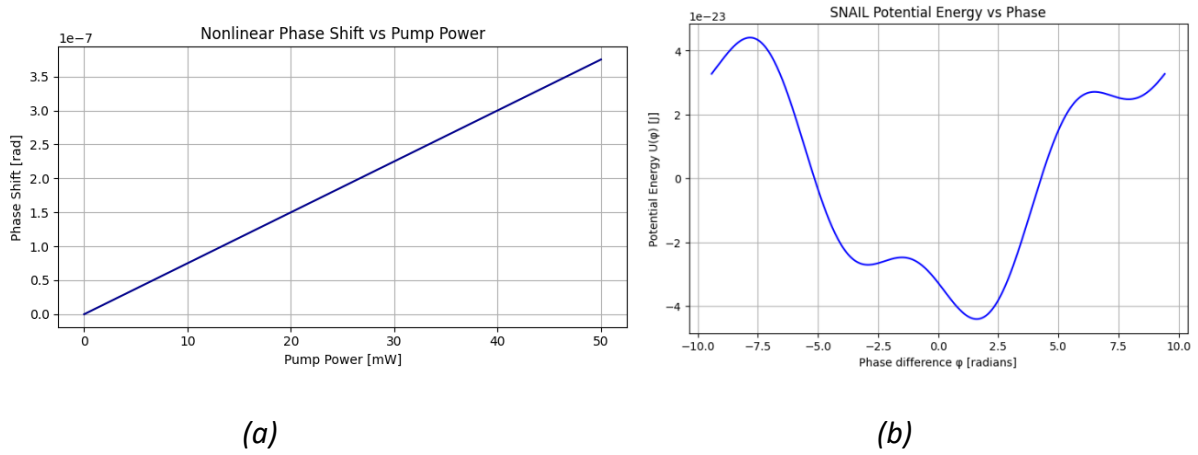


Fig 6: (a) three-wave mixing- increase the pump power, the nonlinear phase shift on the signal increases linearly. You can control how strong the cubic phase gate is just by adjusting the pump power. talks about gate tunability and experimental control. (b) potential energy landscape of Superconducting Nonlinear Asymmetric Inductive element (SNAIL). The curve is asymmetric and shows third-order nonlinearity: it means the SNAIL supports three-wave mixing processes.

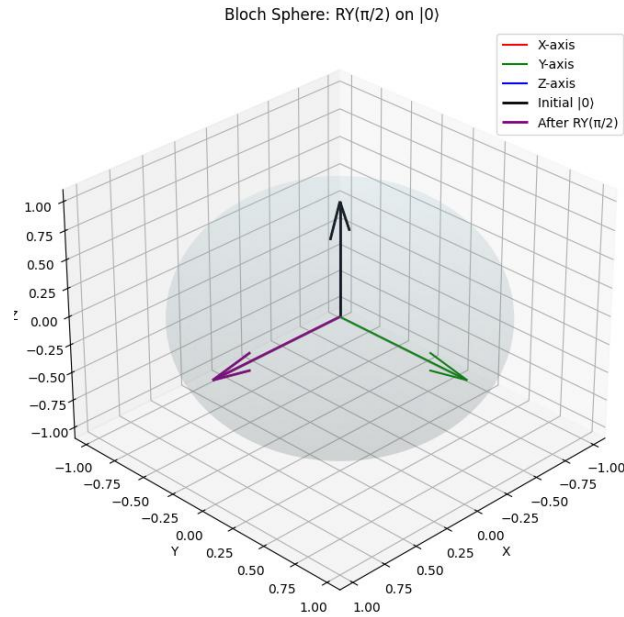


Fig 7: Bloch sphere representation after applying $RY(\pi/2)$ on $|0\rangle$

4. Conclusion

In this report, we explored the theoretical foundations, material considerations, and design strategies necessary for realizing optical non-Clifford gates, with a focus on the cubic phase gate. We highlighted the pivotal role of the cubic phase gate in achieving universal continuous-variable (CV) quantum computation by enabling operations beyond the Gaussian (Clifford) regime. Through a study of three-wave mixing processes in second-order

nonlinear materials such as periodically poled lithium niobate (PPLN), we examined practical methods to engineer the required cubic nonlinearity.

Simulation results demonstrated how a SNAIL (Superconducting Nonlinear Asymmetric Inductive eLement) circuit can induce a third-order nonlinear potential through asymmetry in its energy landscape, supporting three-wave mixing necessary for cubic gate realization. Furthermore, we showed that the nonlinear phase shift increases linearly with pump power, offering a tunable mechanism to control the strength of the cubic phase interaction.

By combining phase rotation gates, three-wave mixing dynamics, and proper material engineering, we presented a feasible scheme for implementing a cubic phase gate using integrated photonic or superconducting platforms. The Bloch sphere visualizations and the Qiskit simulations validated the transformation induced by non-Clifford gates, highlighting their critical importance for universal CV quantum computing.

Overall, while challenges such as loss, phase-matching, and scalability remain, the integration of engineered non-Gaussian operations—such as the cubic phase gate—represents a crucial step toward fault-tolerant, scalable quantum photonic technologies.

6. References

[1] N. Budinger, A. Furusawa, and P. van Loock, "**All-optical quantum computing using cubic phase gates**," *arXiv preprint*, arXiv:2211.09060v2 [quant-ph], 13 Jul 2024.

[2] N. Frattini, "**Three-wave Mixing in Superconducting Circuits: Stabilizing Cats with SNAILS**," *Ph.D. Dissertation*, Yale Graduate School of Arts and Sciences, Fall 2021.

[3] S. Ghose and B. C. Sanders, "**Non-Gaussian Ancilla States for Continuous Variable Quantum Computation via Gaussian Maps**," *Journal of Modern Optics*, vol. 54, no. 6, pp. 855–869, 2007

7. Appendix

```
<ipython-input-5-a537ae55fa48>:40: DeprecationWarning: scipy.misc.derivative
second = derivative(snail_potential, phi_eq, dx=1e-6, n=2, order=5)
<ipython-input-5-a537ae55fa48>:41: DeprecationWarning: scipy.misc.derivative
third = derivative(snail_potential, phi_eq, dx=1e-6, n=3, order=7)
Expansion around minimum (phi_eq = 1.61 rad):
Quadratic coefficient (U2) = 1.372e-23 J
Cubic coefficient (U3)      = -6.612e-21 J/rad
Quartic coefficient (U4)    = -6.906e-14 J/rad^2
<ipython-input-5-a537ae55fa48>:42: DeprecationWarning: scipy.misc.derivative
fourth = derivative(snail_potential, phi_eq, dx=1e-6, n=4, order=9)
```

Cubic phase gate simulation

```
from qiskit import QuantumCircuit
from qiskit.circuit.library import PhaseGate
```

```
qc = QuantumCircuit(1)
qc.h(0)
qc.append(PhaseGate(np.pi/4), [0])
qc.h(0)
qc.draw('mpl')
```

SNAIL Potential energy vs phase

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.misc import derivative
```

```
# SNAIL Parameters
```

```
phi0 = 2.067833848e-15 # Flux quantum [Wb]
```

```
Ej = 20e9 * 6.626e-34 # Josephson energy of large junction [J] (20 GHz)
```

```
alpha = 0.3 # Asymmetry (small junction ratio, typically 0.2-0.4)
```

```
n_junctions = 3 # Number of large junctions
```

```
external_flux = 0.35 * phi0 # Applied DC flux bias [Wb]
```

```
# Create the potential energy function U(phi)
```

```
def snail_potential(phi):
```

```
    """
```

```
    phi: phase difference across the SNAIL [radians]
```

```
    Returns: Potential energy U(phi) [J]
```

```
    """
```

```
    term_large = -n_junctions * Ej * np.cos(phi / n_junctions)
```

```
    term_small = -alpha * n_junctions * Ej * np.cos(phi - 2 * np.pi * external_flux / phi0)
```

```

return term_large + term_small

# Create phase array
phi = np.linspace(-3*np.pi, 3*np.pi, 500)

# Compute potential
U = snail_potential(phi)

# Plot the SNAIL potential
plt.figure(figsize=(8,5))
plt.plot(phi, U, color='blue')
plt.title("SNAIL Potential Energy vs Phase")
plt.xlabel("Phase difference  $\phi$  [radians]")
plt.ylabel("Potential Energy  $U(\phi)$  [J]")
plt.grid(True)
plt.show()

# Find the expansion coefficients: Quadratic, Cubic, Quartic...
def get_expansion_coeffs(phi_eq):
    """Calculate the Taylor expansion coefficients of  $U(\phi)$  about  $\phi_{eq}$ ."""
    second = derivative(snail_potential, phi_eq, dx=1e-6, n=2, order=5)
    third = derivative(snail_potential, phi_eq, dx=1e-6, n=3, order=7)
    fourth = derivative(snail_potential, phi_eq, dx=1e-6, n=4, order=9)
    return second, third, fourth

# Find minimum of potential (equilibrium point)
phi_min = phi[np.argmin(U)]

# Expansion at minimum
U2, U3, U4 = get_expansion_coeffs(phi_min)

```

```

print("Expansion around minimum (phi_eq = {:.2f} rad):".format(phi_min))
print(f"Quadratic coefficient (U2) = {U2:.3e} J")
print(f"Cubic coefficient (U3) = {U3:.3e} J/rad")
print(f"Quartic coefficient (U4) = {U4:.3e} J/rad^2")

```

Nonlinear phase shift vs pump power

```

import numpy as np
import matplotlib.pyplot as plt

# Physical parameters
L = 5e-3 # waveguide length in meters
gamma = 1.5e-3 # effective nonlinearity constant (1/W/m)
P_pump = np.linspace(0, 50e-3, 100) # pump power in W (0–50 mW)

# Input signal field
A0 = 1.0 # signal amplitude (normalized)

# Nonlinear phase shift:  $\phi = \gamma * P_{\text{pump}} * L$ 
phi = gamma * P_pump * L

# Output field with phase modulation
A_out = A0 * np.exp(1j * phi)

# Plot phase vs power
plt.figure(figsize=(7, 4))
plt.plot(P_pump * 1e3, np.unwrap(np.angle(A_out)), color='darkblue')
plt.title("Nonlinear Phase Shift vs Pump Power")
plt.xlabel("Pump Power [mW]")
plt.ylabel("Phase Shift [rad]")
plt.grid(True)

```

```
plt.tight_layout()
```

```
plt.show()
```

