class StabilityAnalysis:

def \_\_init\_\_(self):

self.perturbation\_modes = None

self.background\_solution = None

def analyze\_stability(self, field\_config):

"""

Analyze stability of temporal flow solutions

"""

# Compute Lyapunov exponents

lyapunov = self.compute\_lyapunov\_spectrum(field\_config)

# Analyze perturbation growth

growth\_rates = self.analyze\_perturbations(field\_config)

# Energy conditions

energy\_conditions = self.check\_energy\_conditions(field\_config)

# Validate solution stability

is\_stable = all([

max(lyapunov) < 0, # All modes decay

max(growth\_rates) < 1e-10, # No exponential growth

energy\_conditions # Energy conditions satisfied

])

return is\_stable, {

'lyapunov': lyapunov,

'growth\_rates': growth\_rates,

'energy\_conditions': energy\_conditions

}

def compute\_lyapunov\_spectrum(self, config):

"""

Compute spectrum of Lyapunov exponents to analyze chaos

"""

# Implementation of numerical Lyapunov computation

# Using QR decomposition method

pass

# Usage

stability = StabilityAnalysis()

is\_stable, metrics = stability.analyze\_stability(field\_configuration)