Signals-and-Systems-CA / Coding-Activity-1.ipynb

Sweep76 finalized 09e2042 · 6 minutes ago

100 lines (100 loc) · 213 KB

Coding Activity 1

Joshua Patrick G. Chiu

Using Python (or some other programming language), plot sampled versions of the sinusoidal signal described as:

$$x(t) = 9cos(200\pi t + 0.4\pi)$$

over the range $-0.02 \leq t \leq 0.02$ seconds. For your output, you should produce three graphs with different sample intervals Ts, namely $T_s = 0.0075$, $T_s = 0.005$, and $T_s = 0.0005$.

Solution:

In [2]:

```
import numpy as np
import matplotlib.pyplot as plt

# Define sinusoidal signal
def x(t):
    return 9 * np.cos(200 * np.pi * t + 0.4 * np.pi)

# Define time range & sample intervals
t = np.linspace(-0.02, 0.02, 1000)

# Sampling intervals
Ts = [0.0075, 0.005, 0.0005]

# Colors for the sampled points
colors = ['b', 'g', 'r']  # Blue, Green, Red for different sampling intervals

# Create subplots
fig, axs = plt.subplots(3, 1, figsize=(10, 12))

# Plot for each sample interval
for i, Ts_value in enumerate(Ts):
    t_sampled = np.arange(-0.02, 0.02, Ts_value)
    x_sampled = x(t_sampled)

    # Plot the continuous signal in grey
    axs[i].plot(t, x(t), color='grey', linewidth=0.5, label='Continuous Signal')

    # Plot the sampled points in different colors
    axs[i].stem(t_sampled, x_sampled, basefmt='k-', linefmt=f'{colors[i]}-', mar

    # Set titles and labels
    axs[i].set_title(f'Samples of Sinusoid: $T_s = {Ts_value} s$', color=colors[
    axs[i].set_xlabel('Time $t$ (s)')
    axs[i].set_ylabel('$x(t)$')
    axs[i].grid(True, linestyle=':')
    axs[i].legend()

# Adjust layout
```

```
plt.tight_layout()
plt.show()
```