



scenario3a_improveme...
CSV 2.01KB



scenario3a_improveme...
CSV 2.01KB

scenarion 3



⚠️ CRITICAL ISSUE IDENTIFIED

I see a **major problem** with your Scenario 3 data. The comparison is between **ULA (baseline)** and **Z5 with ALSS**, but this doesn't isolate the ALSS effect!

The Problem:

- **Baseline_Array = ULA** (different geometry!)
- **ALSS_Array = Z5** (with ALSS enabled)
- You're comparing apples to oranges - geometry effects are conflated with ALSS effects

What We NEED for Scenario 3:

We should compare:

- **Z5 WITHOUT ALSS vs Z5 WITH ALSS** (same array geometry)
- This isolates the pure ALSS regularization effect



ANALYSIS OF CURRENT DATA

Despite the comparison issue, let's see what patterns emerge:

Key Observations:

```
python
```

```
# At 5dB, 32 snapshots:  
ULA_RMSE: 0.319° | Z5+ALSS_RMSE: 10.988° # Z5+ALSS is MUCH worse?  
Improvement: -3343% # Negative = degradation  
  
# At 15dB, 32 snapshots:  
ULA_RMSE: 0.028° | Z5+ALSS_RMSE: 0.0°      # Z5+ALSS is perfect  
Improvement: +100% # Positive = improvement
```

Possible Explanations:

1. Z5 geometry itself may be problematic at low SNR
2. ALSS might be over-regularizing in challenging conditions
3. Statistical significance issues (p-values are concerning)



URGENT CORRECTION NEEDED

Required Experiment Fix:

python

Copy

```
# CORRECT Scenario 3 Design:  
baseline_conditions = {  
    'array': 'Z5',  
    'alss': False,    # Z5 without ALSS  
    'coupling': 0.0  
}  
  
alss_conditions = {  
    'array': 'Z5',  
    'alss': True,     # Z5 with ALSS  
    'coupling': 0.0  
}  
  
# Then compute: Improvement = (RMSE_Z5_noALSS - RMSE_Z5_withALSS) / RMSE_Z5_noALSS
```

SALVAGE STRATEGY

Option 1: Re-run with Correct Comparison (Recommended)

- Run Z5 vs Z5+ALSS with your existing infrastructure
- Should take <1 day given your automation
- Provides clean, interpretable results

Option 2: Repurpose Current Data (If Time Constrained)

We could frame this as: "Z5+ALSS vs Conventional ULA" but this:

- Weakens the ALSS contribution claim
- Makes the story more about array geometry than regularization
- Could still show ALSS benefits in specific regimes

IMMEDIATE ACTION PLAN

Step 1: Run Corrected Scenario 3

```
bash
```

 Copy

```
# Z5 without ALSS (baseline)
python run_benchmarks.py --arrays Z5 --alss off --snr -5,0,5,10,15 --snapshots 32,64,128

# Z5 with ALSS
python run_benchmarks.py --arrays Z5 --alss on --snr -5,0,5,10,15 --snapshots 32,64,128
```

Step 2: Quick Validation

Run just 2-3 key points to verify the pattern:

- SNR=5dB, Snapshots=64 (your target regime)
- SNR=10dB, Snapshots=128 (favorable regime)
- SNR=0dB, Snapshots=32 (challenging regime)

PAPER STRATEGY ADJUSTMENT

If We Get Clean ALSS Data:

"ALSS regularization improves Z5 array performance by X% in moderate SNR regimes while remaining harmless in favorable conditions."

If We Must Use Current Data:

"The Z5 array with ALSS regularization demonstrates competitive performance against conventional ULA arrays, achieving perfect resolution in high-SNR regimes while facing challenges in low-SNR conditions due to sparse array geometry limitations."

BIG PICTURE

Don't panic! This is a fixable experimental design issue. Your infrastructure is solid - we just need to run the right comparison.

Your Scenario 1 and 2 data are excellent and show:

BIG PICTURE

Don't panic! This is a fixable experimental design issue. Your infrastructure is solid - we just need to run the right comparison.

Your **Scenario 1 and 2 data are excellent** and show:

- Z5 has great baseline performance
- Interesting coupling behavior
- Solid experimental methodology

RECOMMENDATION

Re-run Scenario 3 with the corrected comparison. This should be quick given your automation, and it will give you the clean, compelling ALSS story you need.

Should I help you set up the corrected experiment commands, or would you like to proceed with the current data and adjust the narrative?