



ΕΛΛΗΝΙΚΟ ΜΕΣΟΓΕΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ – ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

# Υπολογισμός Παραμέτρων Άνω Ζεύξης (Uplink)

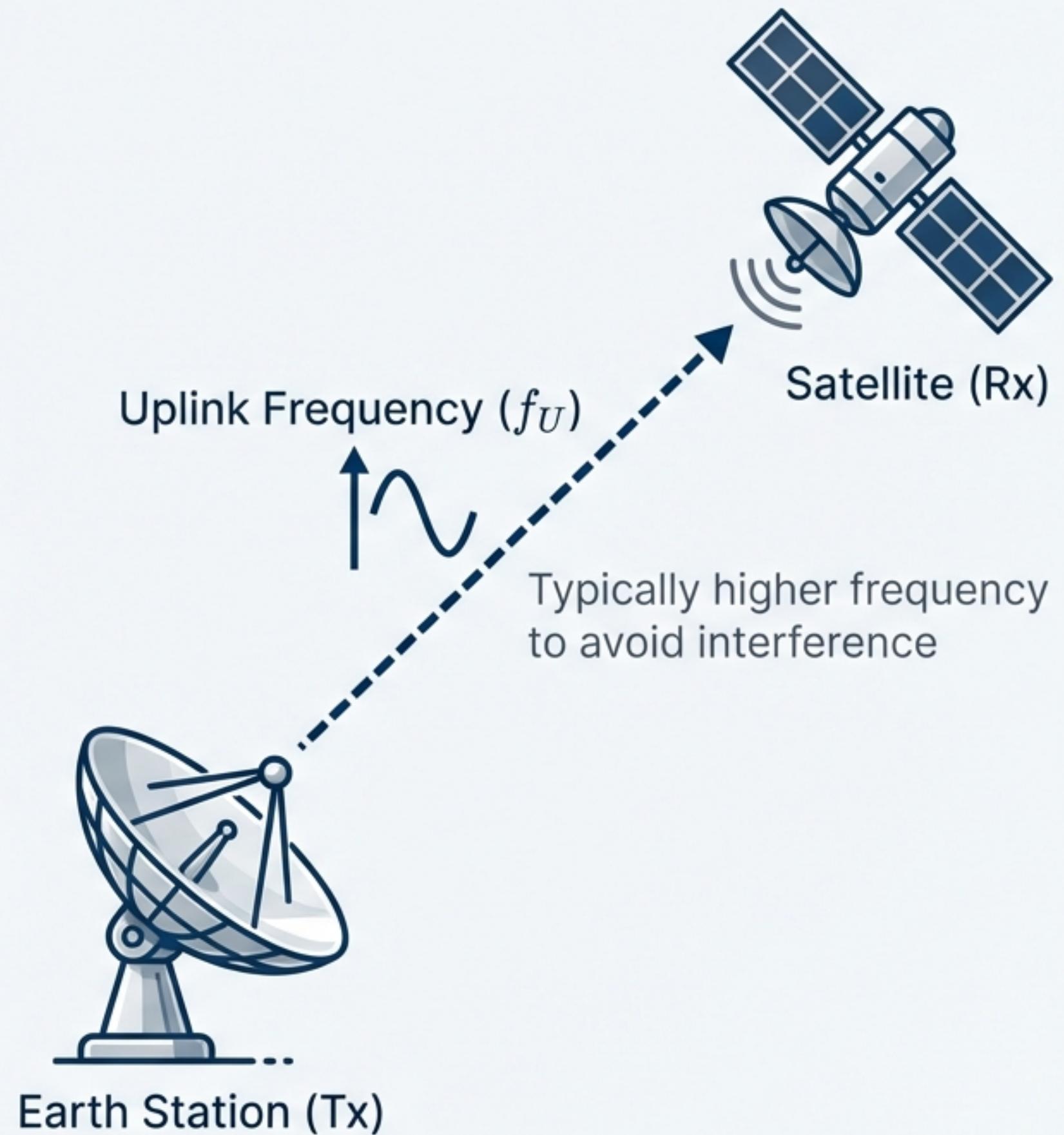


# Εισαγωγή & Σκοπός

Η 'Άνω Ζεύξη' (Uplink) αποτελεί τον ζωτικό δίαυλο επικοινωνίας από τον Επίγειο Σταθμό (Earth Station) προς τον Δορυφόρο.

## Στόχοι:

- **Μελέτη:** Ανάλυση της ηλεκτρομαγνητικής θεωρίας και των απωλειών διαδρομής.
- **Προσομοίωση:** Ανάπτυξη λογισμικού σε Python 3 για τον δυναμικό υπολογισμό του Ισοζυγίου Ζεύξης (Link Budget).
- **Οπτικοποίηση:** Γραφική απεικόνιση σημάτων (QPSK/QAM) και αποτελεσμάτων σε πραγματικό χρόνο.



# Θεωρητικό Υπόβαθρο: Ισοζύγιο Ζεύξης

$$P_R = EIRP + G_R - L_{total}$$

Ισχύς λήψης  
στον δορυφόρο  
(Received Power)

$$P_T - L_{line} + G_T$$

Ενεργός Ισοτροπική  
Ακτινοβολούμενη  
Ισχύς

Κέρδος  
κεραίας λήψης  
δορυφόρου  
(Receive Gain)

Συνολικές Απώλειες:

- FSPL (Ελεύθερος Χώρος)
- Ατμοσφαιρικές  
(Atmospheric)
- Βροχής (Rain Fade)
- Σκόπευσης (Pointing Loss)

Η εξίσωση αυτή αποτελεί τη βάση για τον υπολογισμό του λόγου  $C/N_0$  και  
του **περιθωρίου ζεύξης** (Link Margin).

# Μαθηματικά Μοντέλα & Υλοποίηση

## Free Space Path Loss

$$L_{FS} = 32.45 + 20 \log_{10}(f_{MHz}) + 20 \log_{10}(d_{km})$$

Implementation

```
def fspl_db(f_mhz, d_km):  
    d_km = max(d_km, 1e-9)  
    return 32.45 + 20.0 * math.log10(f_mhz) + 20.0 * math.log10(d_km)
```

## Antenna Gain

$$G_{dBi} = 10 \log_{10} \left[ \eta \left( \frac{\pi D}{\lambda} \right)^2 \right]$$

Implementation

```
def dish_gain_dbi(d_m, f_mhz, eff):  
    # ... (υπολογισμός μήκους κύματος λαμ) ...  
    g_lin = eff * (math.pi * d_m / max(lam, 1e-12)) ** 2  
    return 10.0 * math.log10(max(g_lin, 1e-12))
```

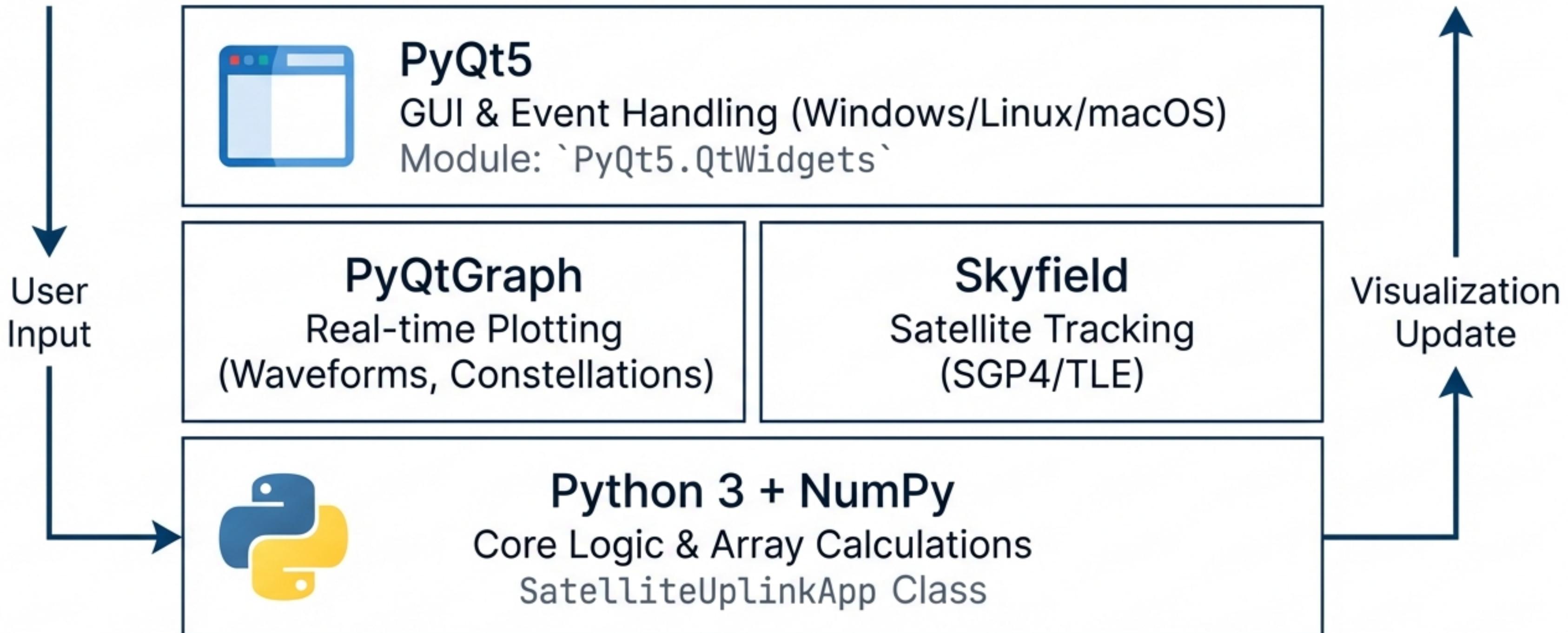
## Pointing Loss

$$L_{point} = 12 \left( \frac{\theta_{error}}{\theta_{3dB}} \right)^2$$

Implementation

```
def pointing_loss_db(mispoint_deg, hpbw_deg_val):  
    ratio = (mispoint_deg / max(hpbw_deg_val, 1e-6))  
    return 12.0 * (ratio ** 2)
```

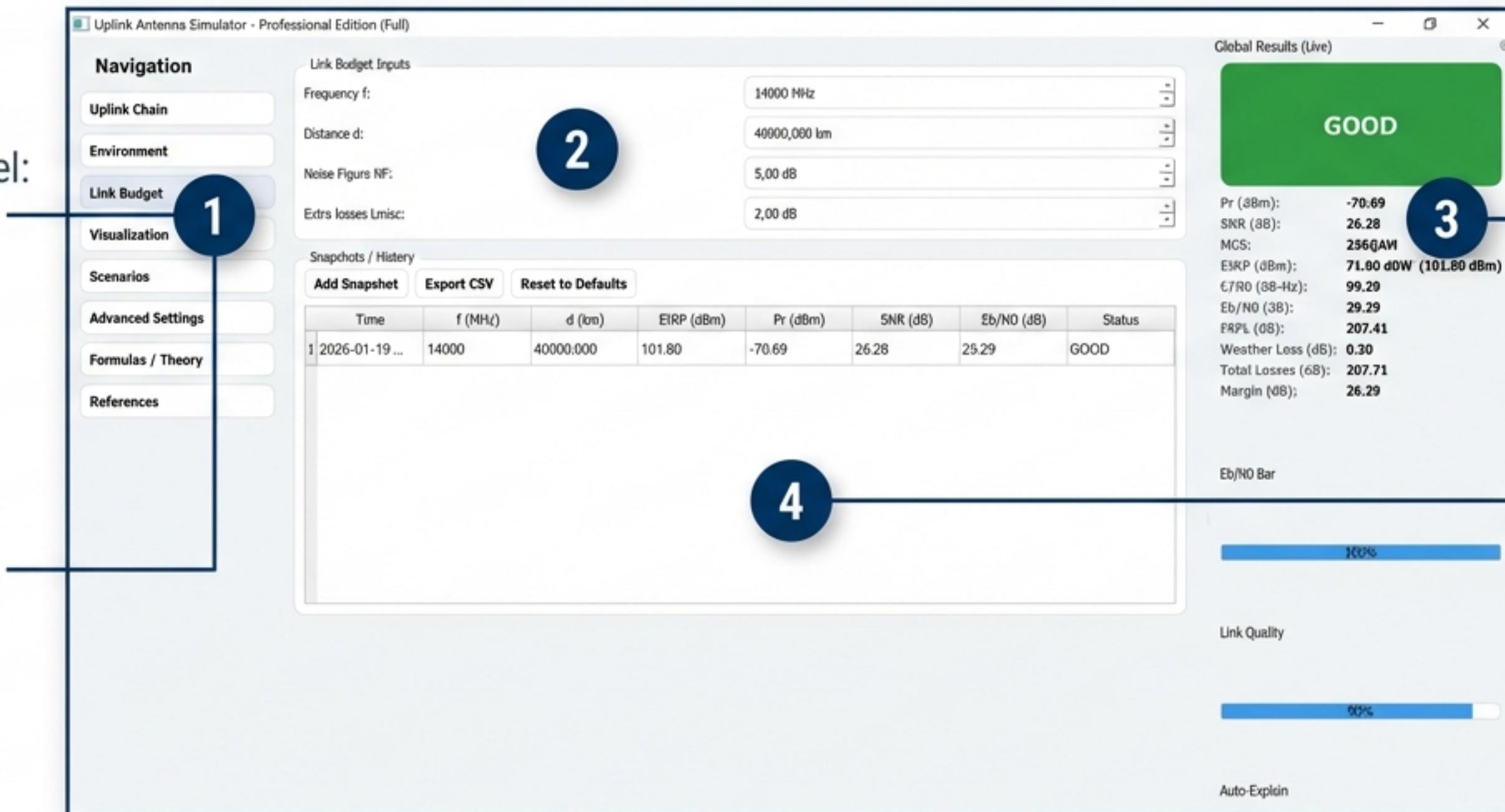
# Αρχιτεκτονική Λογισμικού



`calculate_link_budget() → Math Formulas → update_plots()`

# Περιβάλλον Διεπαφής Χρήστη

1. Navigation Panel:  
Πλοήγηση στις  
καρτέλες.



2. Parameter  
Inputs:  
Εισαγωγή  
 $f, d, P_{tx}, G_{ant}$ .

3. Global Results:  
Ζωντανά  
αποτελέσματα  
( $C/N_0, Status$ ).

4. History:  
Καταγραφή  
στιγμιότυπων.

# Μεθοδολογία Επαλήθευσης

Satellite Theory Mode

The Standard



Βιβλιογραφία: Βουγιούκας, «Ανάλυση και Σχεδίαση Δορυφορικών Ζεύξεων», σελ. 99–102.

Χρήση λυμένων παραδειγμάτων για validation της μηχανής υπολογισμών.

The Equation

$$(C/N_0)_U = \text{EIRP} - L_U + (G/T)_{Sat} + 228.6$$

- $L_U$ : Total Uplink Losses ( $FSPL + L_{atm} + L_{rain} + L_{pol}$ )
- $(G/T)_{Sat}$ : Satellite Figure of Merit
- **228.6**: Constant ( $10\log(1/k)$ )

Διάκριση Μοντέλων: Χρήση ντετερμινιστικών τιμών dB για καιρικές απώλειες (για επαλήθευση τύπων) vs. Στοχαστικών μοντέλων (για δυναμική προσομοίωση).

# Σενάριο 1: Αιθρία (Clear Sky Verification)

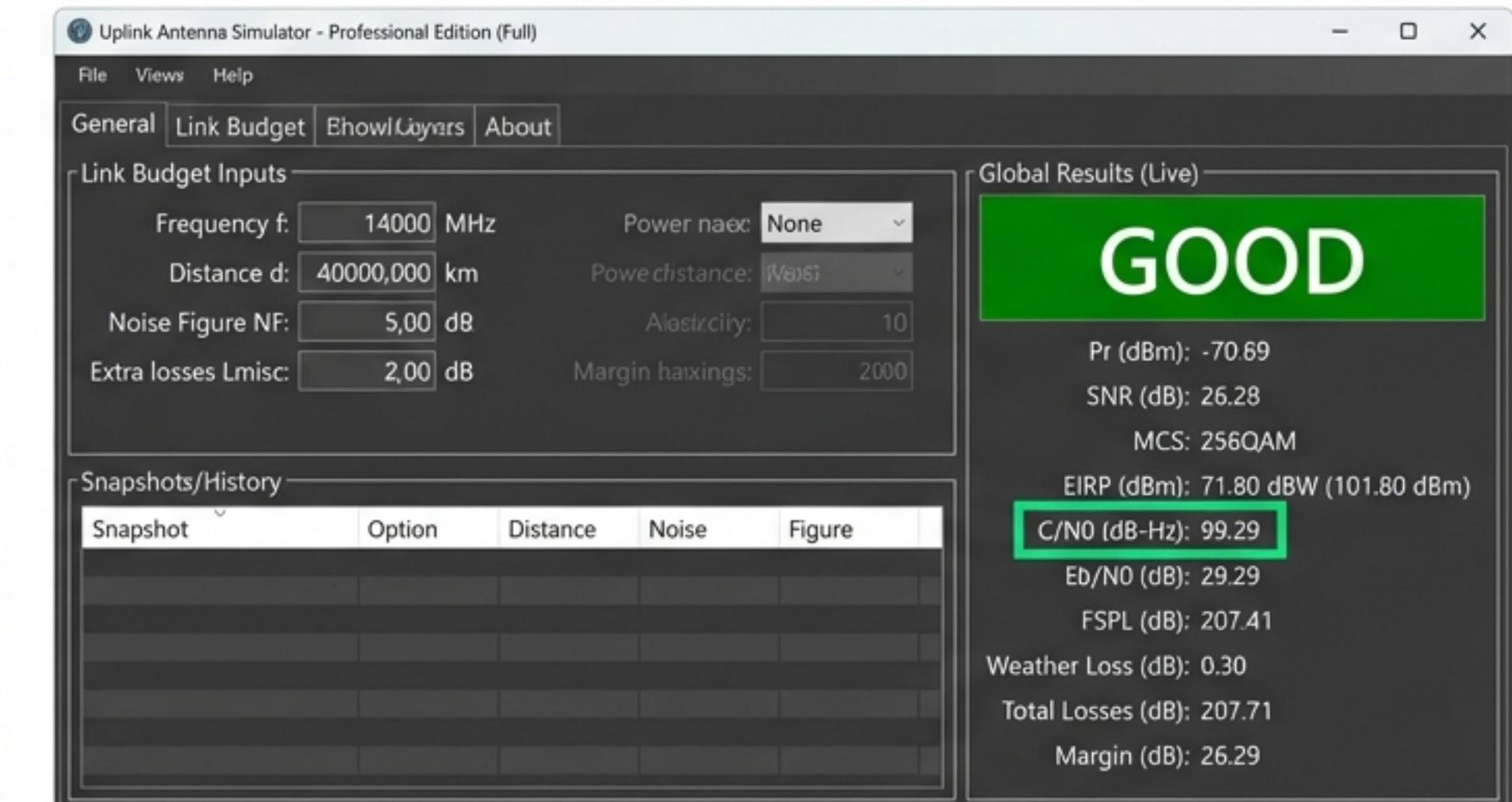
## Theoretical Parameters

Textbook Inputs (Vougioukas, pg. 99)

- $f = 14.0 \text{ GHz}$  (Ku-Band)
- $d = 40,000 \text{ km}$
- $P_{tx} = 50 \text{ dBm}$  (20 dBW)
- $G_{tx} \approx 53.1 \text{ dBi}$  (Dish  $D = 4\text{m}$ )

Expected Result:  $C/N_0 = 99.2 \text{ dBHz}$

## Simulator Output



**STATUS: VERIFIED ( $\Delta < 0.1 \text{ dB}$ )**

# Σενάριο 2: Βροχόπτωση (Rain Fade Verification)

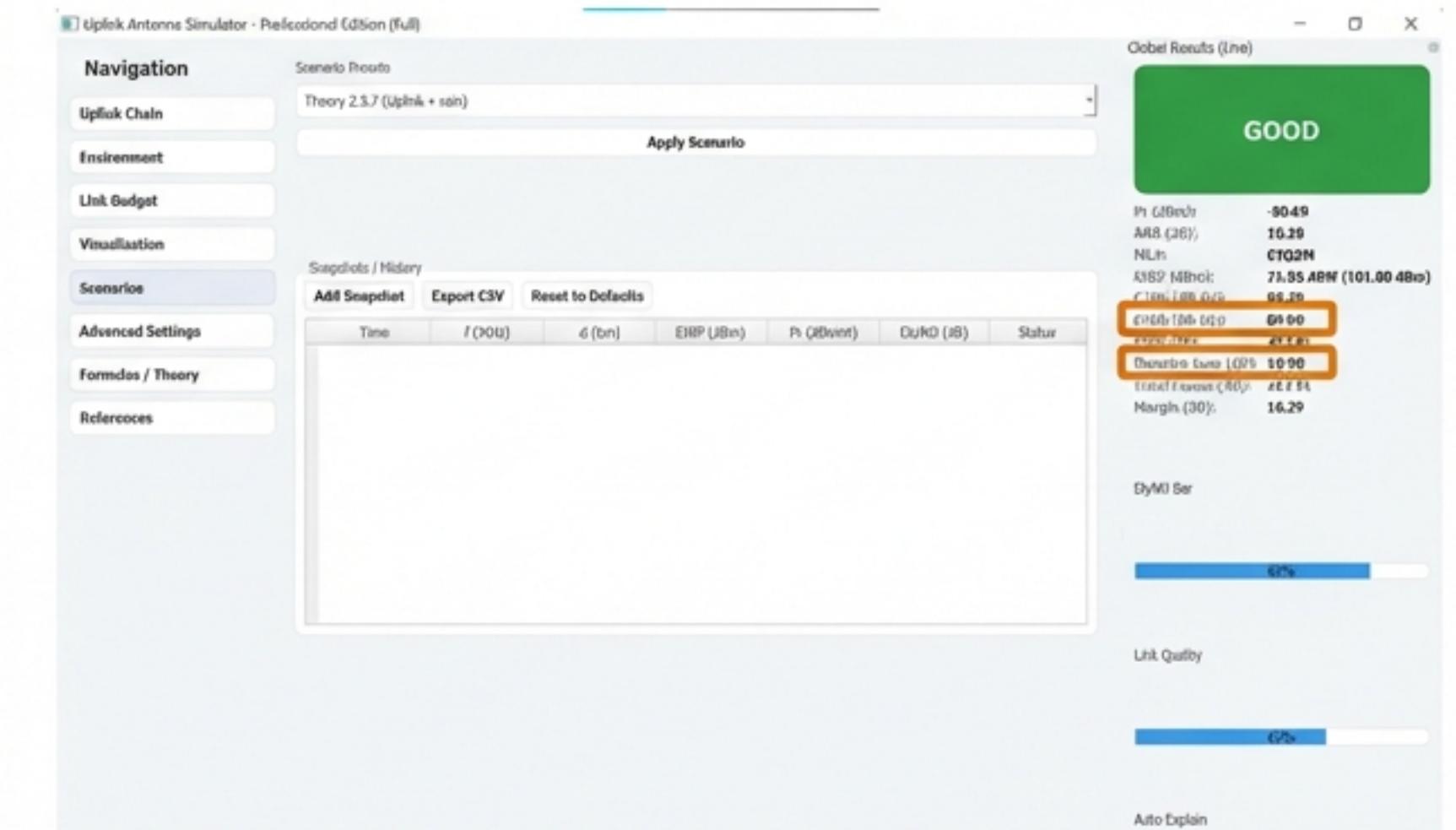
## Theoretical Parameters

### Textbook Inputs (Vougioukas, pg. 102)

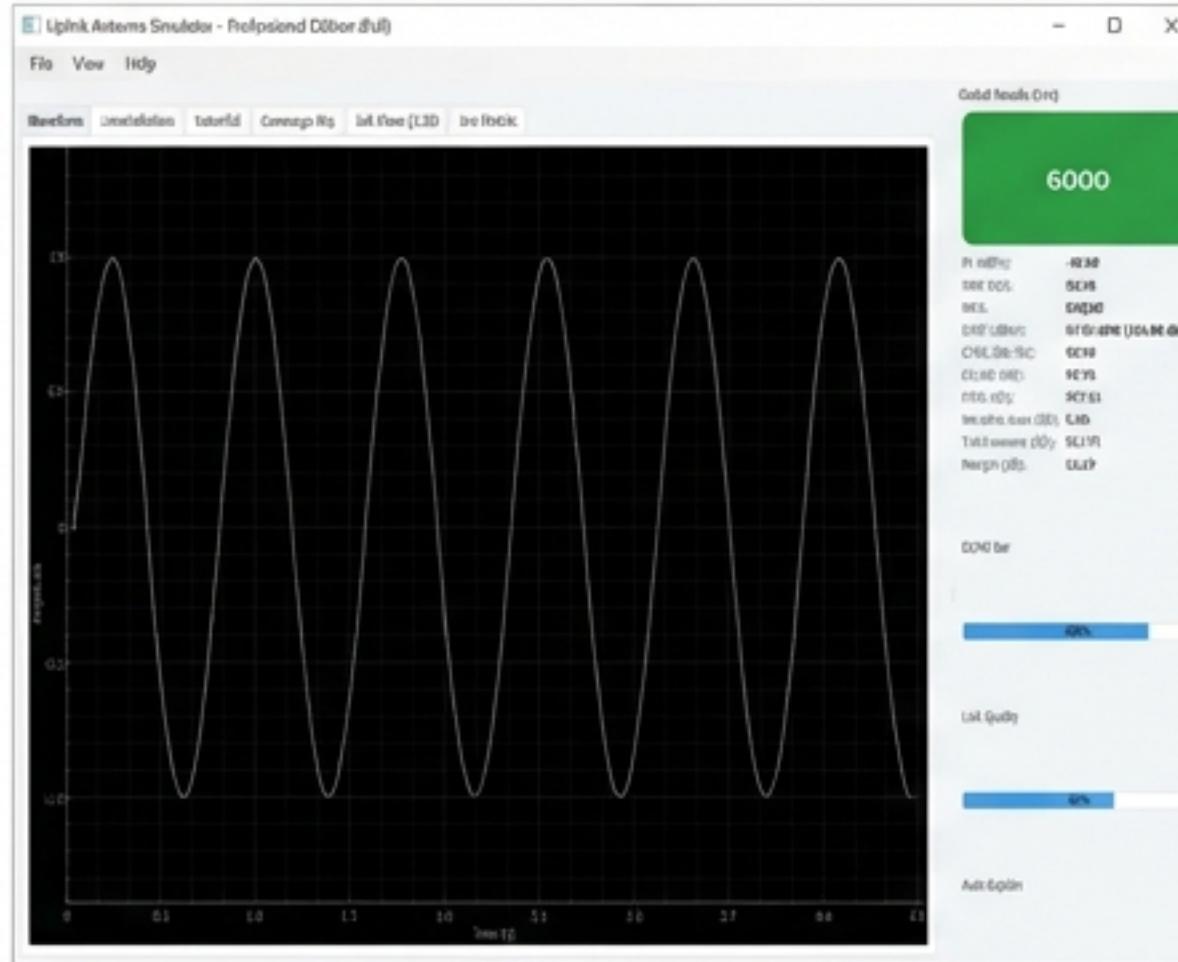
- Condition: Rain Fade
- Added Loss:  $L_{rain} = 10.0 \text{ dB}$
- Other parameters: Same as Scenario 1

Expected Result:  $C/N_0 = 89.2 \text{ dBHz}$

## Simulator Output

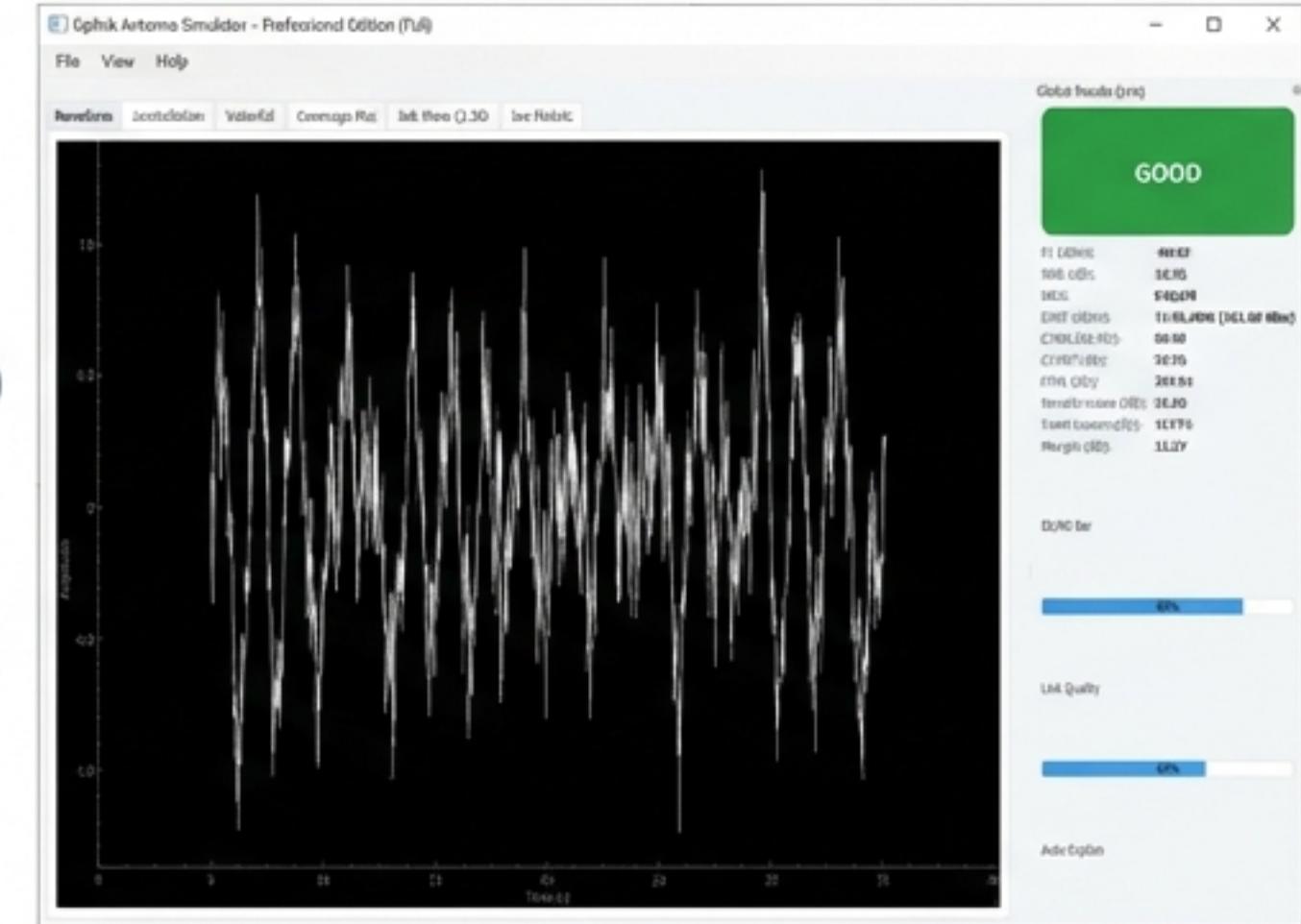


# Οπτικοποίηση Σήματος (Visualization)



Waveform (Time Domain):  
Απεικόνιση πλάτους/χρόνου.

- **Waveform:** Έλεγχος ποιότητας φέροντος.
- **Constellation:** Διάγραμμα αστερισμού (QPSK/QAM) για οπτικό έλεγχο διασποράς συμβόλων.
- **Waterfall:** Οπτικοποίηση κατανομής ισχύος.

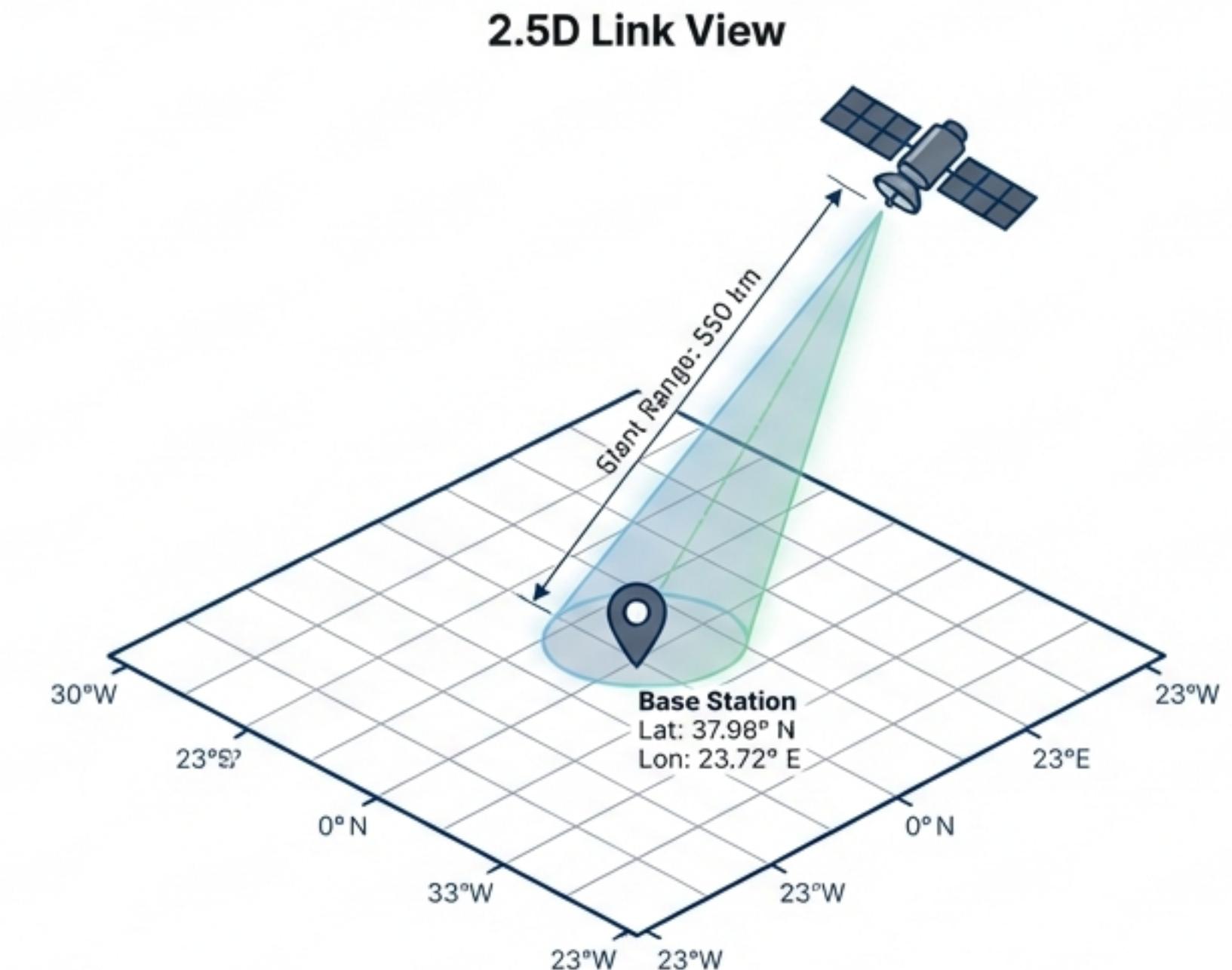


High Noise Scenario: Επίδραση του θορύβου στο σήμα.

# Προηγμένη Ιχνηλάτηση (Advanced Tracking)

Ενσωμάτωση βιβλιοθήκης ‘**Skyfield**’ για τροχιακούς υπολογισμούς.

- **TLE Files:** Φόρτωση πραγματικών δεδομένων τροχιάς (Two-Line Elements).
- **Real-Time Geometry:** Υπολογισμός Slant Range και Elevation βάσει συντεταγμένων GPS.
- **Supported Orbits:** LEO (e.g., Starlink, ISS) και GEO.



# Καταγραφή & Ανάλυση Δεδομένων

- **Live Metrics:** Κυλιόμενα γραφήματα (Rolling plots) για  $P_r$ ,  $SNR$ .
- **Snapshots:** Λειτουργία `Add Snapshot` για σύγκριση σεναρίων.

Snapshots / History								
Add Snapshot		Export CSV		Reset to Defaults				
	Time	f (MHz)	d (km)	EIRP (dBm)	Pr (dBm)	SNR (dB)	Eb/N0 (dB)	
1	2026-01-19 ...	14000	40000.000	101.80	-70.69	26.28	29.29	GOOD
2	2026-01-19 ...	14000	40000.000	101.80	-80.69	16.28	19.29	GOOD

- **Export:** Εξαγωγή σε CSV για περαιτέρω επεξεργασία.

# Τεχνικές Προκλήσεις & Επίλυση

Challenge	Solution
High DPI Scaling (Blurry UI on Windows 10/11)	Override high DPI scaling behavior → System (Enhanced)
Offline Operation (No TLE fetch)	Fallback μηχανισμός σε cached αρχεία ή απενεργοποίηση του Satellite Tracker.
Missing Dependencies (DLL errors)	Εγκατάσταση Microsoft Visual C++ Redistributable.

```
> pip install PyQt5 numpy pyqtgraph skyfield
```

# Συμπεράσματα



**Reliability:** Η ακρίβεια του Link Budget επαληθεύτηκε απόλυτα μέσω των θεωρητικών σεναρίων ( $\Delta \approx 0$ ).



**Physics Insight:** Η μετάβαση από Ku σε Ka band αυξάνει δραματικά την ευαισθησία στη βροχή.



**Tool Value:** Ο συνδυασμός Python και PyQt5 προσφέρει ισχυρή υπολογιστική ισχύ σε φιλικό περιβάλλον.



**Performance:** Η χρήση FEC είναι κρίσιμη για τη διατήρηση της ζεύξης σε χαμηλό  $E_b/N_0$ .

# Βιβλιογραφία & Εγκατάσταση

Για την εκτέλεση του λογισμικού:

```
> pip install PyQt5 numpy pyqtgraph skyfield
```

## References

1. Βουγιούκας, Δ., «Ανάλυση και Σχεδίαση Δορυφορικών Ζεύξεων», Πανεπιστήμιο Αιγαίου.
2. Maral, G., & Bousquet, M., «Satellite Communications Systems», Wiley.
3. ITU-R P.618: Propagation data and prediction methods.
4. Skyfield Documentation: astronomy and orbital mechanics in Python.

