

Diseño Experimental

Plan de medidas

Para este proyecto se ha seleccionado un único escenario de medida de tipo suburbano, que conecta la Escuela de Ingeniería de la UAB con la empresa Sener. El recorrido tiene una longitud aproximada de 1,1 km y atraviesa una zona caracterizada por pequeños valles y, especialmente, por la presencia de la autopista AP-7, que se distingue por un tráfico denso y constante a lo largo del día.

Este escenario se clasifica como suburbano porque se sitúa dentro de un campus universitario, rodeado de zonas verdes y próximo a pequeñas poblaciones. Además, la combinación de áreas abiertas, ligeras elevaciones del terreno y la influencia directa de infraestructuras como la autopista aportan un contexto realista y representativo de los entornos suburbanos actuales.

El diseño de la ruta permite evaluar cómo afectan tanto las características naturales (valles, desniveles) como las artificiales (autopista, tráfico intenso) a la propagación de la señal y al rendimiento del sistema de comunicaciones. Esta elección responde al objetivo de obtener resultados que sean extrapolables a otros entornos suburbanos donde coexisten espacios abiertos, barreras geográficas menores y la presencia de infraestructuras relevantes.



Figure 1. Vista aérea del escenario experimental empleado para la caracterización de la turbulencia atmosférica. La línea roja representa el trayecto del haz óptico entre el edificio de Sener y la Escuela de Ingeniería de la UAB, atravesando las autovías AP-7 y B-30.

Instrumentación

La instrumentación principal empleada en este proyecto ha sido el **scintilómetro BLS450** (Scintec), que consta de una unidad emisora y una receptora, ambas montadas sobre trípodes independientes. El scintilómetro opera transmitiendo radiación electromagnética de intensidad constante a una longitud de onda de 850 nm, a lo largo de una distancia horizontal comprendida entre 500 m y 6000 m [1, 34.3], [2].

Dependiendo de la distancia del trayecto óptico, es necesario configurar los interruptores DIP del receptor y ajustar los modos de operación del emisor, activando las opciones de *long path* o *boost* si la distancia o la atenuación lo requieren, siguiendo las especificaciones del manual del fabricante [2]. El emisor dirige el haz hacia el receptor, mientras que las fluctuaciones turbulentas del aire —debidas principalmente a variaciones de temperatura y humedad— provocan distorsiones en los frentes de onda y, como consecuencia, fluctuaciones en la intensidad de la señal recibida [1]. El receptor detecta estas variaciones y las convierte en información útil sobre la turbulencia atmosférica a lo largo de la trayectoria.

El receptor se conecta a una unidad de procesamiento que se comunica mediante un cable Ethernet con un ordenador portátil donde se ejecuta el software *SRun-Scin 1.68*. Este programa permite monitorizar en tiempo real la alineación y la señal recibida (indicadores XA y XB), garantizando que la señal esté dentro del rango recomendado (habitualmente entre 30,000 y 60,000 cuentas, según manual y configuración) [2]. Solo cuando la señal recibida se mantiene dentro de este rango, los datos obtenidos pueden considerarse válidos para el análisis.

Como el sistema no dispone de sensores meteorológicos integrados, las variables ambientales (temperatura, humedad, presión) se midieron manualmente mediante termómetros e higrómetros digitales y una aplicación de barómetro en el móvil. Las alturas de los edificios se determinaron con un medidor láser.



(a) Fuente de alimentación del sistema, utilizada para proporcionar energía a los equipos de medida.



(b) Receptor del scintilómetro BLS450, montado sobre trípode.



(c) Unidad de adquisición para sensores meteorológicos y conexión de dispositivos al receptor.



(d) Emisor del scintilómetro, encargado de enviar la señal óptica. En la parte trasera se observan las opciones de configuración.



(e) Proceso de ajuste para configurar los interruptores DIP.



(f) Vista del interior del equipo mostrando el conector y los interruptores DIP configurables según el trayecto óptico.

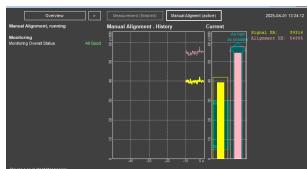


(g) Medidor láser digital utilizado para medir la distancia exacta entre emisor y receptor.

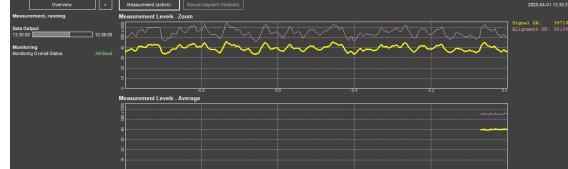


(h) Termohigrómetro digital, empleado para registrar temperatura y humedad durante las medidas.

Figure 2. Instrumentación utilizada en la campaña experimental: (a-d) componentes principales del scintilómetro BLS450; (e-f) detalles de configuración y ajuste del receptor; (g-h) instrumentos auxiliares empleados para la caracterización ambiental y la medición de alturas de los edificios.



(a) Pantalla de alineación manual del software SRun-Scin 1.68, que permite ajustar en tiempo real la orientación del emisor y receptor del scintilómetro optimizando la señal recibida (indicadores XA y XB).



(b) Visualización en el software SRun-Scin 1.68 de los niveles de señal durante la adquisición de datos.



(c) Icónico del software SRun-Scin 1.68, utilizado para la gestión y adquisición de datos en la campaña experimental.



(d) Aplicación móvil de barómetro utilizada para registrar la presión atmosférica en tiempo real durante la campaña de medidas.

Figure 3. Capturas de pantalla y aplicaciones empleadas en el proceso de adquisición y monitorización de datos: (a) alineación manual, (b) monitorización de niveles de señal, (c) acceso al software SRun-Scin 1.68, (d) medición de presión atmosférica con barómetro móvil.

Limitaciones

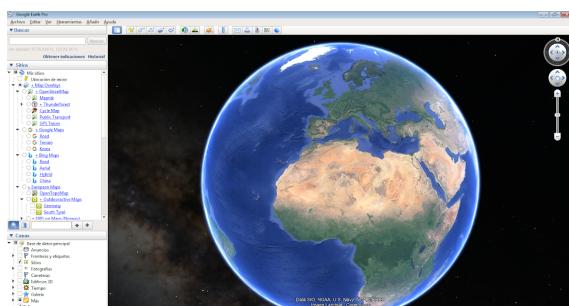


Figure 4. Interfaz principal de la aplicación de Google Earth Pro

Una de las principales limitaciones del presente experimento es la ausencia de una estación meteorológica profesional en el escenario de medida. Como consecuencia, los datos meteorológicos (temperatura, presión y humedad) se recogieron de forma manual mediante aplicaciones móviles, lo que puede introducir desfases temporales respecto a las condiciones reales, especialmente en días con alta variabilidad. Además, se detectaron discrepancias significativas entre diferentes aplicaciones, en particular en la medición de la temperatura, lo que pone en duda la precisión de los sensores integrados en los dispositivos móviles.

Por otro lado, el viento constituye un factor crítico tanto para la propagación óptica como para la estabilidad instrumental. El scintilómetro empleado es especialmente sensible a rachas de viento fuertes, que pueden provocar desalinaciones temporales del haz. Asimismo, el viento puede levantar partículas de polvo o tierra que, al depositarse sobre la óptica del instrumento, afectan a la calidad y continuidad de la señal registrada, incrementando la pérdida de alineación incluso ante pequeñas obstrucciones. Es importante destacar que no se dispuso de un instrumento específico para la medición de la velocidad del viento, lo que limita la capacidad de correlacionar este factor con las posibles pérdidas o perturbaciones observadas durante el experimento.

eaciones temporales del haz. Asimismo, el viento puede levantar partículas de polvo o tierra que, al depositarse sobre la óptica del instrumento, afectan a la calidad y continuidad de la señal registrada, incrementando la pérdida de alineación incluso ante pequeñas obstrucciones. Es importante destacar que no se dispuso de un instrumento específico para la medición de la velocidad del viento, lo que limita la capacidad de correlacionar este factor con las posibles pérdidas o perturbaciones observadas durante el experimento.

Geometría y cálculos

Las distancias del enlace experimental se determinaron utilizando la herramienta de medición de Google Earth Pro. El scintilómetro empleado requiere como parámetro de entrada una única altura efectiva para ambos extremos del enlace; por ello, se calcula la altura media del enlace, siguiendo el procedimiento detallado en el Apéndice ??.

Adicionalmente, para caracterizar la topografía del trayecto, se obtuvo el perfil de elevación del terreno mediante Google Earth, tal como se muestra en la Figura 5. Este perfil permitió identificar los principales desniveles presentes a lo largo de los 1,08 km del enlace UAB-Sener y obtener una representación precisa de la geometría real del haz óptico, además de ser necesario para el cálculo de la altura media del enlace.

Cabe señalar que la distancia proporcionada por Google Earth corresponde a la distancia horizontal entre ambos puntos. Sin embargo, para el enlace óptico es necesario conocer la distancia real (de punto a punto), que puede diferir debido a los desniveles del terreno entre los extremos. El procedimiento para el cálculo de esta distancia efectiva, así como el de la altura media, se detalla en el Apéndice ??.



Figure 5. Perfil de elevación a lo largo del trayecto experimental entre el edificio de Sener y la Escuela de Ingeniería de la UAB. Se observa la variación topográfica en los 1,08 km del enlace, incluyendo el punto de mínima cota y los principales desniveles.

Recogida de datos

Cronograma de medidas

Las campañas de medidas comenzaron el 26 de marzo y se prolongaron hasta el 3 de abril, abarcando inicialmente el trayecto desde Sener hasta la Escuela de Ingeniería (UAB). Durante este período, se planificó la recolección de datos en diversas localizaciones representativas —urbanas, suburbanas, forestales y mixtas—, seleccionadas para analizar el efecto de la turbulencia atmosférica en diferentes entornos, tal y como se recoge en la Tabla 1 y se describe en detalle en el Apéndice ??.

Sin embargo, la campaña experimental se vio condicionada por factores climatológicos adversos, concretamente una inusual temporada de lluvias en Barcelona que impidió la toma de datos durante varios días. A pesar de los esfuerzos por reprogramar, la persistencia de la lluvia limitó las oportunidades de medición, ya que no era posible realizar observaciones en esas condiciones.

Posteriormente, el equipo trató de ampliar el número de escenarios evaluados buscando nuevas ubicaciones. Se presentaron diversas propuestas a la empresa Sener, incluyendo los cinco escenarios recogidos en la Tabla 1. No obstante, todas las propuestas alternativas fueron rechazadas por distintas razones, como la seguridad del aparato, la coincidencia con la temporada de vigilancia en ciertas zonas, o la falta de acceso a edificios privados/restringidos.

Debido a estas limitaciones, se optó por centrar las mediciones en el enlace Sener–UAB, consiguiendo un total de siete registros válidos que constituyen la base del análisis. La planificación inicial contemplaba realizar diez campañas en cinco escenarios distintos durante aproximadamente dos meses y medio, pero diversos factores externos impidieron alcanzar este objetivo.

Table 1. Escenarios propuestos para la campaña de medidas

Escenario	Transmisor (TX)	Receptor (RX)	Distancia (km)
Mixto	Tibidabo	Bunkers Carmel	3,65
Forestal	Torre Vigía Montbau	Puig de la Guardia	3,68
Urbano	Bunkers Carmel	Caixabank Building	4,76
Suburbano	UAB Eng. School	Puig de la Guardia	3,23
Suburbano	UAB Eng. School	Sener	1,10

La planificación completa, así como el cronograma finalmente ejecutado, se muestran en el Apéndice A (Fig. A.1–A.2).

Condiciones meteorológicas y metodología de registro

Al finalizar cada jornada de medidas, los datos registrados mediante el programa SRun-Scin 1.68 se exportaban a un archivo Excel, en el que quedaban reflejadas todas las variables relevantes monitorizadas durante el experimento. Las principales columnas recogidas en este archivo son:

- **timestamp (UTC+01:00):** Fecha y hora local de la medida.
- **timestamp (UTC+02:00):** Fecha y hora en horario UTC+2.
- **Cn2:** Estructura del índice de refracción (turbulencia atmosférica).
- **CT2:** Parámetro de estructura de temperatura.
- **H_convection:** Altura de convección estimada.
- **crosswind:** Velocidad del viento transversal.
- **sigCrosswind:** Desviación estándar del viento transversal.
- **pressure:** Presión atmosférica (hPa).
- **temp:** Temperatura (°C).
- **humidity:** Humedad relativa (%).
- **error:** Indicador de error o calidad del registro.

Adicionalmente, cada día se redactaba un archivo .txt donde, cada veinte minutos, se describían de forma cualitativa las condiciones meteorológicas (presencia de nubes, cobertura solar, viento, etc.). Paralelamente, también se actualizaban manualmente los valores de temperatura, humedad y presión, y se tomaba una fotografía del cielo para documentar visualmente las condiciones de cada intervalo. Se puede consultar el ejemplo de todos estos registros diarios pueden consultarse en el apéndice.

Tratamiento de datos y control de errores

Los datos registrados por el scintilómetro BLS450 fueron exportados en formato digital utilizando el software SRun-Scin 1.68. Antes del análisis, se realizó una inspección visual para identificar posibles registros anómalos debidos a problemas de alineación, desconexiones temporales o interferencias externas. Los intervalos con señal fuera del rango recomendado por el fabricante (por ejemplo, por debajo de 30 000 cuentas) se descartaron del análisis.

Para asegurar la fiabilidad de los resultados, se repitieron las mediciones en diferentes períodos del día y se compararon los valores obtenidos bajo condiciones similares. Finalmente, los resultados se contrastaron con referencias bibliográficas y especificaciones del fabricante para validar la coherencia de los datos.

Clasificación e Integración en SatMuQa

Clasificación de la turbulencia atmosférica

Con el fin de seleccionar, de manera objetiva, los escenarios más benignos y más críticos para un enlace óptico en espacio libre, se desarrolló un *script* en Python que analiza y clasifica un conjunto de perfiles de turbulencia atmosférica (C_n^2). El procedimiento se articula en tres etapas:

1. **Cálculo de métricas múltiples.** A cada perfil se le asignó un vector de indicadores que cubre diferentes perspectivas:
 - *Físicas*: Varianza de Rytov σ_R^2 , que cuantifica la intensidad del centelleo.
 - *Estadísticas*: media, desviación estándar, asimetría (*skewness*) y curtosis, describiendo la distribución de C_n^2 .
 - *Estructurales*: volatilidad del perfil (gradientes bruscos) y rango dinámico, que recogen la heterogeneidad vertical de la turbulencia.
2. **Ranking individual por métrica.** Cada tanda se ordenó de peor a mejor según cada indicador anterior, lo que permite detectar qué tandas resultan extremas en un aspecto concreto (p. ej. máxima σ_R^2 , mayor variabilidad, etc.).
3. **Puntuación agregada y selección de escenarios.** Mediante una función de puntuación se identificaron las tandas que aparecen con mayor frecuencia en los extremos (mejores o peores) de todas las métricas. Ello conduce a dos casos de referencia:
 - **Escenario peor caso**: perfil más severo, idóneo para pruebas de estrés.
 - **Escenario mejor caso**: perfil más benigno, utilizado como punto de referencia bajo condiciones casi ideales.

El *script* produce automáticamente tres salidas:

- **Tabla de métricas** mostrada en consola, con todos los valores calculados por tanda.
- **Ranking final** basado en la puntuación agregada.
- **Informe PDF** que detalla las clasificaciones individuales y ofrece una explicación interpretativa de cada métrica.

Además, los perfiles quedan etiquetados de forma categórica (*baja*, *media* o *alta* turbulencia), facilitando su uso posterior en simulaciones y análisis comparativos.

Las metricas aplicadas se pueden ver en el apendice.

Estructura de entrada para SatQuMA

El software *SatQuMA* v2.0.0 modela las pérdidas del canal atmosférico mediante *suites* de transferencia radiativa (p. ej. MODTRAN o libradtran) o bien la lectura de ficheros de eficiencia de enlace, pero carece de un modelo nativo que traduzca el índice de estructura de la turbulencia C_n^2 en la atenuación óptica correspondiente.. Para suplir esta carencia y realizar un análisis más granular se ha desarrollado un módulo personalizado que integra los efectos de la turbulencia en la simulación.

Este módulo se ha implementado en la subcarpeta `channel/Turbulence`, respetando la arquitectura modular de *SatQuMA*.

El primer paso consiste en procesar los datos de C_n^2 recogidos por un scintilómetro.

Para ello, el script `build_cn2_csv.py` consolida las mediciones de múltiples ficheros Excel en un único archivo maestro, `cn2_data.csv`, que contiene una serie temporal del índice de estructura de la turbulencia C_n^2 :

```
timestamp,Cn2
2025-03-26 12:32:00,1.6927e-14
2025-03-26 12:34:00,1.0495e-14
2025-03-26 12:36:00,1.8170e-14
2025-03-26 12:42:00,7.6740e-14
```

2025-03-26 12:44:00,3.7303e-14

2025-03-26 12:46:00,3.4734e-14

Modelado de perfiles de sobrevuelo a partir de series temporales

SatQuMA evalúa el rendimiento del enlace durante un sobrevuelo, discretizado según el ángulo de elevación del satélite.

Para simular el efecto de la turbulencia se debe establecer una correspondencia entre la serie temporal de C_n^2 medida in situ y la variabilidad esperada a lo largo de un sobrevuelo completo ($0^\circ - 90^\circ$).

Segmentación temporal de C_n^2 y construcción de perfiles de sobrevuelo

Se adopta la hipótesis de que la evolución temporal de la turbulencia atmosférica puede emplearse como un *proxy* para describir su variabilidad espacial y angular. Para implementar este enfoque, el script `cn2_data.py` segmenta la serie temporal contenida en el archivo `cn2_data.csv` en bloques consecutivos de 91 muestras.

Cada bloque se interpreta como un *perfil de sobrevuelo* potencial:

- Cada una de las 91 mediciones dentro del bloque se asigna a un ángulo de elevación entero, que varía desde 0° hasta 90° .
- De esta manera, se obtiene un perfil que simula la evolución del C_n^2 durante un paso completo del satélite, cubriendo todas las elevaciones relevantes del enlace.

Este procedimiento permite generar perfiles de turbulencia representativos de un sobrevuelo típico, facilitando el análisis y la simulación de escenarios en comunicaciones ópticas en espacio libre.

El procedimiento automatizado es el siguiente:

1. Cargar el archivo maestro `cn2_data.csv`.
2. Agrupar las mediciones por día de observación.
3. Para cada día, segmentar la serie temporal en *tandas*, esto es, bloques de 91 muestras.
4. Exportar cada tanda a un fichero CSV independiente
(p. ej. `2025-03-26_tanda1.csv`), preservando la estructura `timestamp, Cn2`.

Este método genera un conjunto diverso de perfiles de turbulencia estandarizados, listos para convertirse en el formato que el simulador puede interpretar.

Generación de un módulo de datos de turbulencia y extensión del simulador

El paso final del pre-procesamiento es la creación de un módulo Python, `pasosCn2.py`, que actúa como base de datos autocontenido de perfiles de turbulencia. El script `pasosCn2_builder.py` automatiza esta tarea: lee todos los ficheros de tandas `*.csv`, los organiza en una matriz de `NumPy` y genera el código Python del módulo. Dicho módulo expone los perfiles de C_n^2 junto con sus etiquetas correspondientes (p. ej. `tanda3_14.05`), de modo que puedan importarse y seleccionarse fácilmente desde el simulador principal.

Para que SatQuMA consuma esta nueva fuente de datos fueron necesarias dos modificaciones clave en su código fuente.

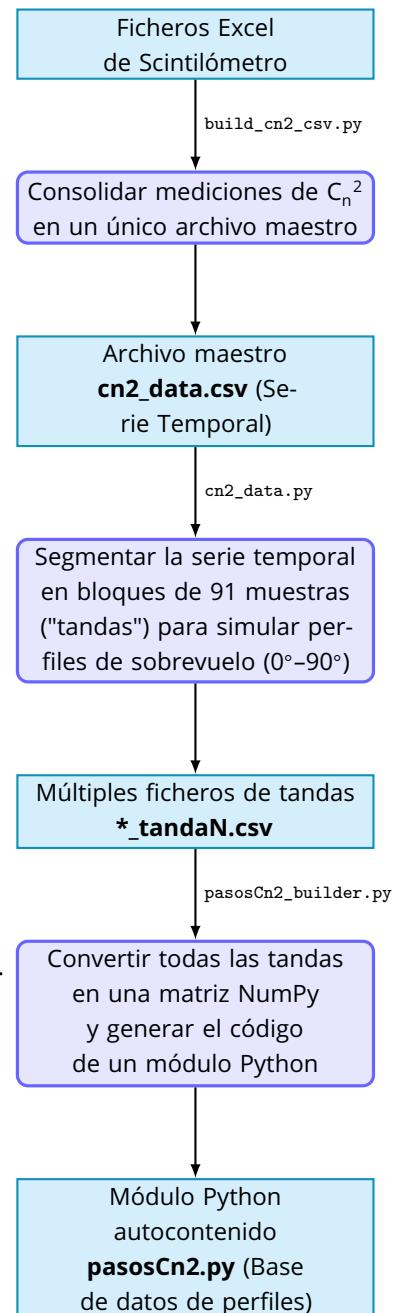


Figure 6. Diagrama de flujo del pipeline de pre-procesamiento para generar perfiles de turbulencia a partir de datos brutos.

Extensión del fichero de entrada

Se añadieron cuatro parámetros al archivo de configuración `input.txt`, con los que el usuario controla la simulación de turbulencia:

- `turbulence`: bandera booleana (True/False) que activa o desactiva el modelo de pérdidas por turbulencia.
- `tanda_label`: etiqueta del perfil de C_n^2 a emplear (debe coincidir con una definida en `pasosCn2.py`).
- `wavelength`: longitud de onda del sistema (en metros).
- `L`: longitud total del enlace óptico (en metros).

Estos parámetros se leen al inicio de la simulación y determinan si se invoca la lógica de turbulencia.

Modelo Computacional de la Varianza de Rytov

Esta sección detalla la justificación teórica y la estrategia computacional implementada para modelar el efecto de la turbulencia atmosférica en los enlaces ópticos satelitales. El objetivo es construir una función que describa la varianza de Rytov, σ^2 , de manera precisa y computacionalmente eficiente.

Fundamento Físico y Modelo Atmosférico

El principal efecto adverso de la turbulencia atmosférica en un haz láser es la fluctuación de su irradiancia, fenómeno conocido como centelleo (*scintillation*). La magnitud de este efecto se cuantifica mediante la **varianza de Rytov** (σ^2), una métrica adimensional que, para una onda plana, se define por la ecuación:

$$\sigma^2 = 1.23 \cdot C_n^2 \cdot k^{7/6} \cdot L^{11/6} \quad (1)$$

donde:

- C_n^2 es el coeficiente de la estructura del índice de refracción atmosférico, que cuantifica la intensidad de la turbulencia.
- k es el número de onda de la luz, definido como $k = 2\pi/\lambda$.
- L es la longitud del camino óptico a través del medio turbulento.

En el contexto de un enlace satélite-tierra, la turbulencia atmosférica se concentra principalmente en las capas más bajas de la atmósfera. Por encima de una cierta altitud, la densidad del aire es tan baja que los valores de temperatura, humedad y presión resultan en un C_n^2 prácticamente despreciable. Por ello, aunque la distancia total al satélite pueda ser de cientos de kilómetros, solo el tramo que atraviesa la atmósfera densa contribuye significativamente al centelleo.

En este modelo, se define un parámetro clave: la **altura efectiva de la turbulencia** (H_{turb}). Este valor representa el espesor de la capa atmosférica turbulenta que el haz debe atravesar. Para esta simulación, se ha adoptado un valor de referencia de $H_{turb} = 80$ km, aunque este parámetro es configurable para permitir el análisis de diferentes condiciones atmosféricas.

El factor geométrico más dinámico en un enlace satelital es la longitud del camino turbulento, L . Este valor no es constante, sino que presenta una fuerte dependencia del **ángulo de elevación del satélite** (θ). La relación geométrica se puede aproximar como:

$$L(\theta) \approx \frac{H_{turb}}{\sin(\theta)}$$

Esta dependencia implica una conclusión fundamental: la varianza de Rytov no es un valor estático, sino una **función continua del ángulo de elevación**, $\sigma^2(\theta)$. El objetivo primordial del módulo de turbulencia es, por tanto, modelar esta función de manera robusta y eficiente.

Algorithm 1 Generador de la Función de Varianza de Rytov: `get_f_sigma2`

Variables Globales Persistentes:

`_sigma2_cache`: Diccionario (caché) para almacenar funciones de interpolación ya generadas.
`tandas, elev_deg, cn2_table`: Datos de perfiles de turbulencia predefinidos.

```
1: function get_f_sigma2(loss_params)
2:   if loss_params['turbulencia'] es False then
3:     return una función  $\lambda x : 0.0$                                  $\triangleright$  Se anula el efecto si la turbulencia está desactivada.
4:   end if
5:   Extraer tanda_label, wl, H_turb de loss_params.
6:   if alguno de los parámetros anteriores es nulo then
7:     Lanzar error de configuración.
8:   end if
9:   Crear clave de caché: cache_key  $\leftarrow$  (tanda_label, wl, H_turb).
10:  if cache_key existe en _sigma2_cache then
11:    return _sigma2_cache[cache_key]                          $\triangleright$  Devolver función pre-calculada desde el caché.
12:  end if                                                  $\triangleright — \text{Inicio del cálculo (solo si no está en caché)} —$ 
13:  Calcular número de onda:  $k \leftarrow 2\pi/\text{wl}$ .
14:  Convertir vector de elevación a radianes: elev_rad  $\leftarrow$  np.radians(elev_deg).
15:  Calcular vector de longitud de camino turbulento: L_path  $\leftarrow$  H_turb / sin(elev_rad).
16:  Obtener índice idx correspondiente a tanda_label.
17:  Extraer el perfil de turbulencia: cn2_vals  $\leftarrow$  cn2_table[:, idx].
18:  Calcular el vector de varianza de Rytov (operación vectorizada):
19:   $\sigma_{\text{vals}}^2 \leftarrow 1.23 \cdot k^{7/6} \cdot \text{cn2_vals} \cdot \text{L}_{\text{path}}^{11/6}$ .
20:  Crear una función de interpolación lineal  $f_{\sigma^2}$  a partir de los puntos (elev_rad,  $\sigma_{\text{vals}}^2$ ).
Configuración: kind='linear', bounds_error=False, fill_value='extrapolate'.
21:  Almacenar la nueva función en el caché: _sigma2_cache[cache_key]  $\leftarrow f_{\sigma^2}$ .
22:  return  $f_{\sigma^2}$ 
23: end function
```

Estrategia Computacional y Algoritmo

Evaluar la Ecuación 1 en cada instante de tiempo de una simulación sería computacionalmente prohibitivo. Para superar este desafío, se ha diseñado una estrategia basada en tres pilares: pre-cálculo vectorizado, interpolación sobre la magnitud física final y memoización (caché). Esta estrategia se materializa en el algoritmo `get_f_sigma2`, descrito en el Algoritmo 1.

La lógica del algoritmo se fundamenta en las siguientes optimizaciones:

1. **Pre-cálculo y Vectorización:** En lugar de realizar cálculos iterativos, el módulo aprovecha las capacidades de cómputo vectorizado de NumPy para calcular la tabla completa de valores de σ^2 para todos los ángulos de elevación disponibles (e.g., de 0° a 90°). Esta operación, aunque intensiva, se ejecuta una única vez por configuración, desacoplando el cálculo físico del bucle principal del simulador.
2. **Interpolación sobre la Magnitud Física Final:** El resultado del pre-cálculo es un conjunto discreto de puntos (θ_i, σ_i^2) . Para obtener valores en ángulos intermedios (p. ej., 25.5°), se recurre a la interpolación. En lugar de interpolar un parámetro de entrada como C_n^2 para luego aplicarlo a la Ecuación 1 (que es no lineal), se adopta un enfoque teóricamente más robusto: se interpola directamente sobre la magnitud física final, σ^2 . Mediante la función `scipy.interpolate.interp_1d`, se construye un objeto de interpolación ligero y eficiente, $f_{\sigma^2}(\theta)$, que proporciona valores de σ^2 de forma casi instantánea.
3. **Memoización (Caché) para la Reutilización:** Para optimizar simulaciones que involucran múltiples análisis con parámetros idénticos (misma tanda de C_n^2 , misma longitud de onda y misma H_{turb}), se implementa un

sistema de caché. Antes de iniciar cualquier cálculo, el algoritmo verifica si ya ha generado una función de interpolación para la misma configuración. Si existe, la recupera de la memoria (`_sigma2_cache`), eliminando por completo la redundancia computacional en ejecuciones sucesivas.

En síntesis, este diseño modular transforma un problema físico-matemático complejo en una herramienta computacional ligera y eficiente. La estrategia de **pre-calcular, interpolar y cachear** permite integrar los efectos de la turbulencia de manera robusta y físicamente coherente, sin penalizar el rendimiento global de la simulación.

Modelo Estocástico del Canal y Muestreo de la Transmisividad

El módulo `get_f_sigma2`, descrito en la sección anterior, proporciona el valor esperado de la varianza de Rytov (σ_R^2) para un ángulo de elevación determinado. Sin embargo, σ_R^2 no es una atenuación directa, sino un **parámetro estadístico** que describe la severidad de las fluctuaciones de irradiancia. La naturaleza de la turbulencia es intrínsecamente aleatoria; en cualquier instante, la transmisividad del canal (T) puede ser mayor o menor que su valor medio debido al centelleo.

El propósito de este segundo módulo es, por tanto, dar el paso fundamental desde un parámetro determinista (σ_R^2) a una **realización estocástica** del estado del canal. Para ello, se implementa un modelo que genera una muestra aleatoria y físicamente plausible de la transmisividad T , utilizando la distribución de probabilidad que mejor describe el comportamiento del haz láser en función del régimen de turbulencia.

Fundamento Teórico: Modelos de Distribución de Irradiancia

La elección del modelo estadístico es crucial, ya que la física de la propagación de la luz a través de la atmósfera cambia drásticamente con la intensidad de la turbulencia. Se distinguen principalmente dos regímenes, cada uno con su modelo de distribución de probabilidad asociado:

1. **Régimen de Fluctuaciones Débiles** ($\sigma_R^2 < 1.0$): En este régimen, el haz láser mantiene en gran medida su coherencia espacial. Las perturbaciones inducidas por la atmósfera son pequeñas y, bajo la teoría de Rytov, se puede demostrar que la irradiancia sigue con gran precisión una **distribución Log-normal**.
2. **Régimen de Fluctuaciones Fuertes** ($\sigma_R^2 \geq 1.0$): El haz sufre una ruptura (*beam breakup*) en múltiples filamentos, y la distribución de la irradiancia se vuelve más compleja. Para este régimen, se ha adoptado el **modelo Gamma-Gamma**. Este modelo postula que la irradiancia es el producto de dos variables aleatorias Gamma independientes, cuyos parámetros α y β se calculan a partir de σ_R^2 . Físicamente, estas variables representan las contribuciones de las fluctuaciones a pequeña y gran escala.

El módulo implementado actúa, por tanto, como un selector de modelo adaptativo: evalúa el valor de σ_R^2 para determinar el régimen físico y, a continuación, muestrea la transmisividad instantánea T de la distribución de probabilidad correspondiente. El proceso se formaliza en el Algoritmo 2.

Algorithm 2 Muestreo de Transmisividad Instantánea: `get_turbulence_for_angle`

```

1: function get_turbulence_for_angle(elevdeg, loss_params)
2:   Obtener la función de interpolación  $f_{\sigma^2}$  llamando a get_f_sigma2(loss_params).
3:   if la obtención de  $f_{\sigma^2}$  falla then
4:     Lanzar error o devolver None.
5:   end if
6:   Convertir ángulo de entrada a radianes:  $elev_{rad} \leftarrow \text{radians}(elev_{deg})$ .
7:   Obtener la varianza de Rylov para el ángulo actual:  $\sigma_R^2 \leftarrow f_{\sigma^2}(elev_{rad})$ .
8:   Muestrear la transmisividad  $T$  llamando a la función interna _sample_transmissivity(\sigma_R^2).
9:   return  $\sigma_R^2, T$                                           $\triangleright$  Devolver la varianza y la muestra de transmisividad
10: end function

11: function _sample_transmissivity(\sigma_R^2)                                 $\triangleright$  Función interna para la selección del modelo
12:   if  $\sigma_R^2 \leq 0$  then
13:     return 1.0                                          $\triangleright$  No hay fluctuaciones sin turbulencia
14:   end if
15:   if  $\sigma_R^2 < 1.0$  then                                 $\triangleright$  Régimen débil: Modelo Log-normal
16:     Calcular parámetros  $\mu, s$  de la distribución Log-normal a partir de  $\sigma_R^2$ .
17:     return una muestra aleatoria de  $\text{LogNormal}(s, \text{scale} = e^\mu)$ .
18:   else                                                  $\triangleright$  Régimen fuerte: Modelo Gamma-Gamma
19:     Calcular parámetros  $\alpha, \beta$  del modelo Gamma-Gamma a partir de  $\sigma_R^2$ .
20:     if los parámetros  $\alpha, \beta$  no son válidos (e.g., infinitos) then
21:       return 0.0                                          $\triangleright$  Desvanecimiento completo de la señal
22:     end if
23:     Muestrear  $X \sim \text{Gamma}(\alpha, 1/\alpha)$  y  $Y \sim \text{Gamma}(\beta, 1/\beta)$ .
24:     return  $X \cdot Y$ 
25:   end if
26: end function

```

Implementación Detallada del Modelo Gamma-Gamma y Conversión a Pérdida

El modelo Gamma-Gamma, necesario para el régimen de turbulencia fuerte, requiere una implementación más detallada. A continuación, se describe la teoría detrás de su implementación y el proceso para convertir la transmisividad muestreada en una pérdida de canal.

El Modelo Físico Gamma-Gamma

El modelo Gamma-Gamma postula que la irradiancia normalizada (I) de un haz óptico en condiciones de turbulencia fuerte es el producto de dos variables aleatorias independientes, I_x e I_y , cada una siguiendo una distribución Gamma:

$$I = I_x \cdot I_y \quad (2)$$

Físicamente, I_x modela las fluctuaciones a gran escala, mientras que I_y modela las fluctuaciones a pequeña escala. La distribución de probabilidad de una variable Gamma se define por sus parámetros de forma (α o β) y escala.

Cálculo de los Parámetros α y β

Los parámetros de forma α y β no son arbitrarios, sino que están directamente relacionados con la física de la atmósfera a través de la varianza de Rylov (σ_R^2). Se calculan:

$$\alpha = \left[\exp\left(\frac{0.49\sigma_R^2}{(1 + 1.11\sigma_R^{12/5})^{7/6}} \right) - 1 \right]^{-1} \quad (3)$$

$$\beta = \left[\exp\left(\frac{0.51\sigma_R^2}{(1 + 0.69\sigma_R^{12/5})^{5/6}} \right) - 1 \right]^{-1} \quad (4)$$

La implementación en el código incluye robustez para casos extremos. Se manejan los errores numéricos que pueden ocurrir cuando σ_R^2 es muy grande (régimen de saturación) y se verifica la validez de los denominadores en las Ecuaciones 3 y 4. Si un denominador es no positivo, implica que el parámetro correspondiente tiende a infinito, lo que físicamente representa una ruptura total de la coherencia del haz. En tal caso, el código asigna una transmisividad de cero.

Muestreo de la Transmisividad

Una vez calculados α y β , la función `_sample_transmissivity` genera la muestra de la transmisividad instantánea T (análoga a la irradiancia normalizada I):

1. Se genera una muestra aleatoria X de una distribución Gamma con parámetro de forma α y escala $1/\alpha$.
2. Se genera una muestra aleatoria Y de una distribución Gamma con parámetro de forma β y escala $1/\beta$.
3. La transmisividad instantánea es el producto de estas dos muestras: $T = X \cdot Y$.

Este proceso constituye una realización de un método de Monte Carlo, donde se genera una instancia aleatoria del estado del canal que es consistente con la severidad de la turbulencia descrita por σ_R^2 .

Conversión de Transmisividad a Pérdida en Decibelios (dB)

El resultado final del muestreo es la transmisividad T , una magnitud adimensional entre 0 y ∞ (aunque normalizada en la media a 1). Un valor de $T = 1$ representa una transmisión sin cambio respecto a la media, mientras que $T \rightarrow 0$ representa un desvanecimiento completo. En comunicaciones, es estándar expresar la atenuación en decibelios (dB). La conversión de la transmisividad T a una pérdida de canal L_{dB} se realiza mediante:

$$L_{\text{dB}} = -10 \log_{10}(T) \quad (5)$$

El signo negativo es crucial, ya que una transmisividad baja ($T < 1$) debe corresponder a una pérdida positiva (atenuación), y una transmisividad alta ($T > 1$) a una ganancia o "pérdida negativa" (mejora instantánea). Por ejemplo:

- Si $T = 0.5$ (50% de la potencia media llega), la pérdida es $L_{\text{dB}} \approx 3.01 \text{ dB}$.
- Si $T = 0.1$ (10% de la potencia media llega), la pérdida es $L_{\text{dB}} = 10 \text{ dB}$.
- Si $T = 2.0$ (el doble de la potencia media llega), la "pérdida" es $L_{\text{dB}} \approx -3.01 \text{ dB}$.

En el simulador, esta pérdida L_{dB} se suma a las otras fuentes de atenuación para obtener la pérdida total del enlace en un instante dado.

Integración del Modelo de Turbulencia en el Bucle de Simulación

Para incorporar los efectos estocásticos de la turbulencia, fue necesario modificar esta función central. Las modificaciones no solo añaden la nueva física, sino que también se diseñaron para mantener la eficiencia computacional y la flexibilidad del simulador. A continuación, se detallan los cambios clave implementados.

Modificación de la Interfaz de la Función

El primer cambio se realizó en la firma de la función para permitir el paso de parámetros de turbulencia de una manera escalable.

- **Versión Original:** `time_dependent_losses(..., f_atm, eta_int)`
- **Versión Modificada:** `time_dependent_losses(..., f_atm, eta_int, loss_params)`

Se ha añadido el argumento `loss_params`, un diccionario de Python que encapsula todos los parámetros de configuración del modelo de pérdidas (e.g., `tanda_label`, `wavelength`, `H_turb`). Este patrón de diseño mejora la legibilidad y facilita la adición de nuevos parámetros en el futuro sin necesidad de alterar la lista de argumentos de la función.

Integración de la Lógica de Turbulencia en el Bucle Principal

El cambio más significativo ocurre dentro del bucle que itera sobre cada instante del sobrevuelo. La lógica se ha optimizado para minimizar la sobrecarga computacional.

- Inicialización Eficiente:** Antes de entrar en el bucle principal, se inicializa una bandera booleana para determinar si el modelo de turbulencia está activado:

```
turbulence_enabled = loss_params.get('turbulence', False)
```

Esta consulta se realiza una única vez para evitar el coste de acceder y verificar el diccionario `loss_params` en cada una de las miles de iteraciones del bucle.

- Llamada Condicional al Modelo Estocástico:** Dentro del bucle, si la bandera `turbulence_enabled` es `True`, se invoca la función principal del modelo estocástico para el ángulo de elevación actual:

```
sigma_R2, T_turb = get_turbulence_for_angle(elev_deg, loss_params)
```

Esta llamada obtiene tanto la varianza de Rytov (σ_R^2) como una muestra aleatoria de la transmisividad instantánea (T_{turb}). Si la turbulencia está desactivada, se asignan valores neutros ($\sigma_R^2 = 0.0$, $T_{\text{turb}} = 1.0$) para que no afecten al cálculo de pérdidas, garantizando la retrocompatibilidad.

Actualización del Cálculo de la Eficiencia Total

La integración física de la turbulencia se refleja en la actualización de la fórmula de la eficiencia total del enlace (η_{tot}).

- Versión Original:** La eficiencia total era el producto de componentes deterministas:

$$\eta_{\text{tot}} = \eta_{\text{atm}} \cdot \eta_{\text{diff}} \cdot \eta_{\text{int}} \quad (6)$$

- Versión Modificada:** La nueva eficiencia total ahora incluye el factor de transmisividad aleatorio debido al centelleo, convirtiendo a η_{tot} en una variable estocástica:

$$\eta_{\text{tot}} = \eta_{\text{atm}} \cdot \eta_{\text{diff}} \cdot \eta_{\text{int}} \cdot T_{\text{turb}} \quad (7)$$

De esta manera, la simulación captura de forma realista las fluctuaciones del canal inducidas por la turbulencia.

Ampliación de la Estructura de Datos de Salida

Finalmente, para permitir un análisis y depuración exhaustivos de los efectos de la turbulencia, la estructura de datos de salida se ha ampliado.

- Versión Original:** El array de salida `vals` contenía 7 columnas con datos orbitales y de eficiencias deterministas.

- Versión Modificada:** El array `vals` ahora tiene 9 columnas. Las dos nuevas columnas almacenan los valores clave generados por el modelo de turbulencia en cada instante:

- **Columna 8 (sigma2):** La varianza de Rytov (σ_R^2).
- **Columna 9 (T_turb):** La muestra de transmisividad estocástica (T_{turb}).

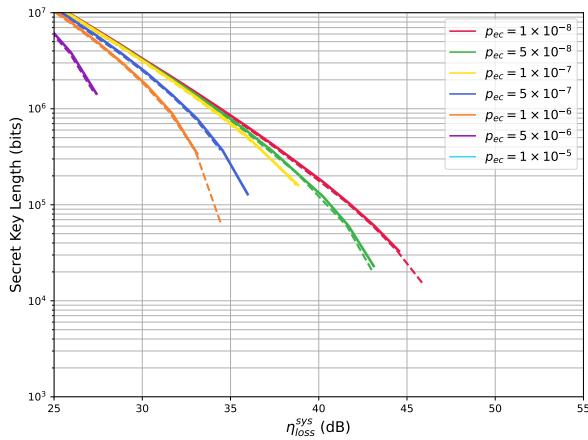
Esta ampliación es crucial para el análisis posterior, ya que permite visualizar cómo evolucionan σ_R^2 y T_{turb} a lo largo del sobrevuelo y correlacionarlos con las métricas de rendimiento del enlace.

Resultados Y ANÁLISIS DE RESULTADOS

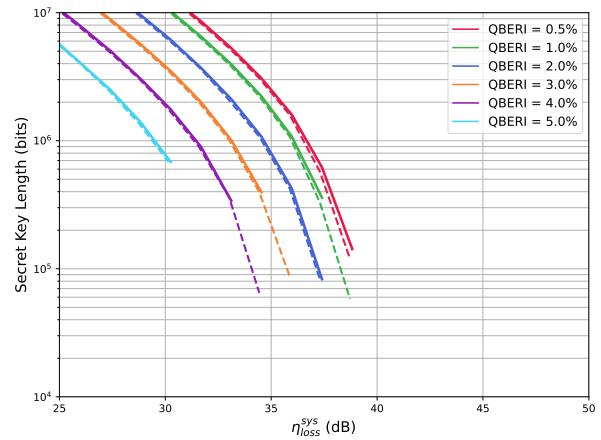
He realizado el análisis utilizando 25 tandas. Para cada tanda, se han considerado 7 valores diferentes de Pec y 5 de QBERI, lo que genera un total de 875 combinaciones posibles. Analizar cada una de estas combinaciones para todos los tipos de gráficos haría que el número total de estudios se disparase exponencialmente.

Para abordar este problema, desarrollamos un script que selecciona aleatoriamente tres tandas para un valor aleatorio de Pec y otro de QBERI. De este modo, se obtiene una muestra representativa sin necesidad de analizar la totalidad de combinaciones.

A pesar de ello, se han realizado numerosos análisis de resultados adicionales para asegurar la robustez y fiabilidad de las conclusiones.



(a) Dependencia de la SKL con Pec, considerando un QBERI de 0,4%. Las líneas continuas representan el sistema sin turbulencia; las líneas discontinuas representan el sistema con turbulencia (tanda1_14.05).



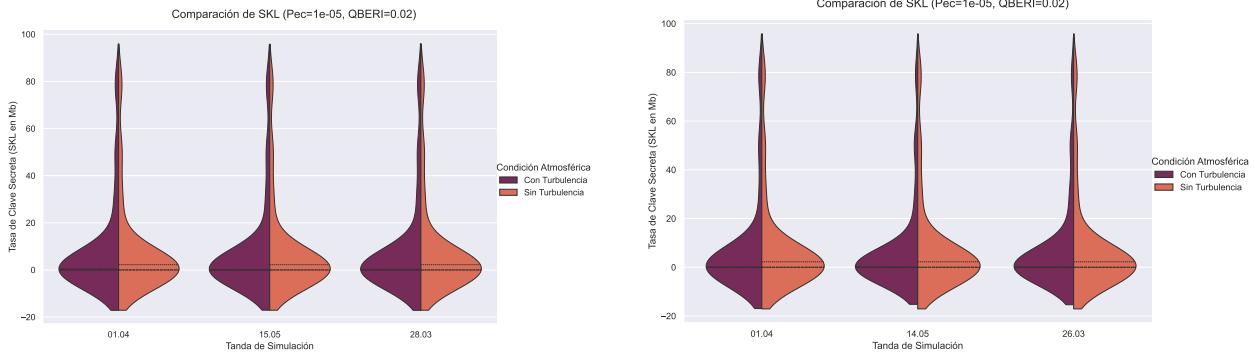
(b) Dependencia de la SKL con el QBERI, considerando un Pec de 1×10^{-6} por pulso. Las líneas continuas representan el sistema sin turbulencia; las líneas discontinuas representan el sistema con turbulencia (tanda1_14.05).

Figure 7. Longitud de clave secreta (SKL) en función de las pérdidas totales del sistema. Los parámetros fijos del sistema son $p_{ap} = 10^{-3}$ y $\theta_{min} = 10^\circ$.

Al analizar la gráfica, observamos que con un valor alto de Pec (probabilidad de conteo espurio), la cantidad de claves secretas generadas disminuye significativamente, dificultando superar pérdidas superiores a 30 dB. Esto se debe a que un Pec elevado incrementa el ruido en el sistema, lo que enmascara la señal útil y reduce drásticamente la tasa de generación de claves.

Por otro lado, al incorporar la variable T_turb en nuestras simulaciones, la cual modela la turbulencia atmosférica, se observa que la mayoría de las tandas con turbulencia experimentan pérdidas promedio más altas. No obstante, pueden generar una cantidad similar de claves secretas si se cuenta con sistemas intrínsecamente mejores (con menores pérdidas internas). La turbulencia introduce además una variabilidad considerable en la señal: aunque pueden existir breves intervalos favorables, predominan los períodos de desvanecimiento profundo, afectando significativamente la generación total de claves secretas.

En el caso específico de un Pec de 1e-06, inicialmente las curvas con y sin turbulencia coinciden. Sin embargo, al aumentar las pérdidas, la curva sin turbulencia muestra una reducción notable en la generación de claves. Al reducir aún más el valor de Pec, el sistema es capaz de generar claves para niveles de pérdida más elevados, llegando hasta aproximadamente 45 dB en las condiciones óptimas de Pec y QBERI. En condiciones menos óptimas, aunque el número de claves generadas es menor, inicialmente las curvas con y sin turbulencia se mantienen cercanas, pero con pérdidas crecientes, la turbulencia provoca una mayor disminución en la generación de claves secretas.



(a) Muestra la comparación usando la tanda 4 del día 01.04, correspondiente aproximadamente a la franja horaria de 10:00 a 12:00; la tanda 2 del día 15.05, correspondiente al periodo de 11:00 a 13:00; y la tanda 2 del día 28.03, también de 10:00 a 12:00.

(b) Compara la tanda 1 del día 26.03 entre las 12:30 y 14:34, la tanda 4 del día 14.05 entre las 14:30 y 15:50, y la tanda 2 del día 15.05 de 11:00 a 13:00.

Figure 8. Comparación de SKL con diferentes condiciones atmosféricas. La comparación general considera un Pec de 1×10^{-7} y un QBER de 0.04, destacando cómo la turbulencia influye en la generación de SKL al comparar diferentes tandas con distintos niveles de turbulencia y también frente a condiciones sin turbulencia.

La turbulencia atmosférica es un factor determinante que degrada el rendimiento de la QKD (Quantum Key Distribution) satelital, afectando significativamente la Tasa de Generación de Clave Secreta (SKL). Su principal efecto es una reducción de la SKL promedio y una mayor variabilidad en su generación en comparación con un escenario ideal sin turbulencia. Esto se debe a las fluctuaciones en la transmisividad del canal (T_{turb}), que provocan desvanecimientos significativos y generan distribuciones de SKL con colas pronunciadas hacia valores bajos o nulos.

El factor más crítico que determina el impacto de la turbulencia es su magnitud global, cuantificada por la varianza de Rytov, por encima de la brusquedad o rapidez de sus fluctuaciones. Por ejemplo, la tanda de simulación del 26.03, caracterizada por una turbulencia intensa y sostenida, exhibió la mayor dispersión en los valores de SKL. En contraste, otras tandas con cambios rápidos pero de menor magnitud mostraron una variación mínima respecto a la condición sin turbulencia.

La sensibilidad del sistema a la turbulencia está fuertemente modulada por sus parámetros operativos, principalmente el error cuántico de bit intrínseco (QBER) y el parámetro de eficiencia del protocolo (Pec):

- Con Pec y QBER bajos, el sistema es robusto y la generación de SKL es muy similar tanto en condiciones con y sin turbulencia.
- Sin embargo, al aumentar Pec y QBER, el sistema se vuelve mucho más vulnerable al ruido y a los errores. Esto amplifica drásticamente el impacto negativo de las fluctuaciones turbulentas, resultando en una reducción drástica de la SKL, que llega a anularse con mayor frecuencia.

Cabe destacar una observación particular: en momentos puntuales, la turbulencia puede causar potenciamientos momentáneos de la transmisividad ($T_{\text{turb}} > 1$), permitiendo que la SKL supere brevemente a la del modelo estático sin turbulencia. No obstante, este efecto es esporádico y no compensa el impacto negativo general causado por los desvanecimientos.

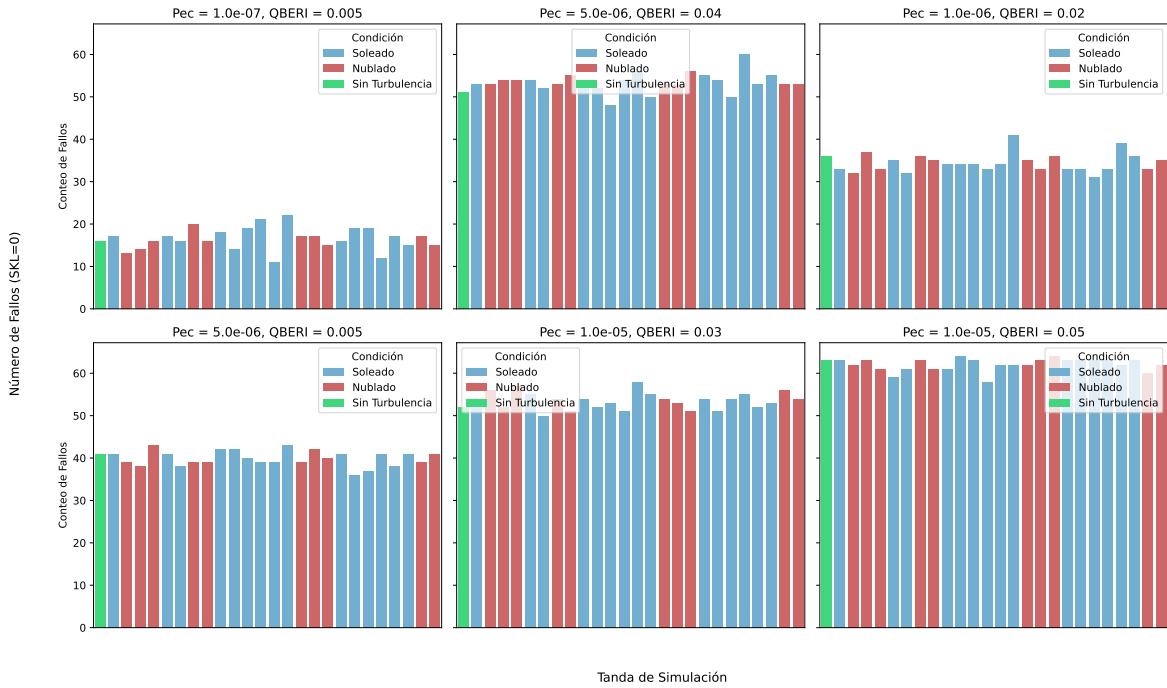


Figure 9. Distribución del número de ejecuciones fallidas ($SKL = 0$) en diferentes tandas y escenarios de turbulencia, ordenadas por hora del día. El histograma muestra el conteo de ejecuciones en las que no se generó clave secreta ($SKL = 0$), agrupadas por tanda y clasificadas según el régimen atmosférico: **Soleado**, **Nublado**. Las tandas están ordenadas cronológicamente, lo que permite analizar la evolución temporal de la probabilidad de fallo a lo largo del día.

Cada fichero de resultados se compone de múltiples *tandas*, que representan ejecuciones independientes del protocolo BB84. Cada fila de un fichero corresponde a una única ejecución bajo condiciones específicas y constantes de hora del enlace, pérdidas ópticas y turbulencia instantánea, a la que se aplica la cadena completa de post-procesado: *sifting*, corrección de errores y *privacy amplification*. La columna **SKL (b)** registra la longitud final de la clave secreta (*Secret Key Length*) obtenida en dicha ejecución, medida en bits.

Cuando las condiciones operativas —principalmente las pérdidas totales, la probabilidad de conteo oscuro (P_{ec}) y la tasa de error cuántico intrínseco ($QBER_i$)— exceden los umbrales de seguridad que el protocolo impone, la clave potencial se descarta. En estos casos, se registra una longitud de clave nula, $SKL = 0$. Por consiguiente, el recuento de filas con $SKL = 0$ dentro de una tanda proporciona una estimación directa de la probabilidad de fallo del enlace, P_{falla} . Un mayor número de ceros implica una menor fiabilidad del sistema bajo ese conjunto de condiciones.

A continuación, se analiza el comportamiento del sistema en diferentes regímenes operativos.

Régimen de alta fidelidad (P_{ec} bajo y $QBER_i$ bajo):

En este escenario, el umbral de error tolerado es mínimo y el ruido intrínseco del sistema (generado por conteos oscuros o defectos ópticos) es prácticamente despreciable. Bajo estas circunstancias, un fallo en la generación de clave ($SKL = 0$) se puede atribuir con alta fiabilidad a las fluctuaciones de transmitancia inducidas por la turbulencia atmosférica. Como resultado, la diferencia de rendimiento entre un día **Soleado** (turbulencia intensa) y uno **Nublado** (turbulencia moderada) se manifiesta con máxima claridad: las tandas con un coeficiente de turbulencia C_n^2 más elevado acumulan un número significativamente mayor de ceros. Empíricamente, estos picos de fallos ($SKL = 0$) se concentran en las horas de mediodía y primera tarde, coincidiendo con el máximo calentamiento solar que intensifica los gradientes térmicos y, con ello, la varianza de Rytov. La simulación de referencia *Sin turbulencia* actúa como una cota inferior, mostrando una tasa de fallos casi nula.

Régimen de operación al límite (P_{ec} bajo y QBER_i alto):

En este caso, el ruido de fondo sigue siendo despreciable, por lo que la tasa de error cuántico global (QBER_{total}) es casi idéntica a la intrínseca (QBER_i). Dado que el protocolo BB84 tolera un máximo teórico del 11 % de error, un QBER_i del 4-5 % ya sitúa al sistema en el límite de su umbral de seguridad. Las cotas estadísticas de Chernoff, utilizadas en la estimación de parámetros, introducen fluctuaciones en cada tanda. Así, una ejecución puede estimar un QBER_{total} = 9.5 % (generando clave, SKL > 0), mientras que la siguiente, bajo idénticas condiciones físicas, podría arrojar un QBER_{total} = 10.1 % (fallando, SKL = 0). Esta extrema sensibilidad a las fluctuaciones estadísticas evidencia la precariedad de operar en un punto tan crítico.

Régimen dominado por ruido de detector (P_{ec} alto y QBER_i bajo):

Aquí, la señal cuántica es intrínsecamente limpia, pero queda enmascarada por la alta tasa de conteos oscuros de los detectores. El método de estados *decoy*, que se basa en comparar las tasas de detección para pulsos de diferente intensidad (señal, *decoy* y vacío), pierde su efectividad al no poder distinguir la señal genuina del ruido. Esto impide acotar correctamente la contribución de los fotones únicos y su correspondiente tasa de error de fase. El resultado es un fallo sistemático del protocolo: incluso sin turbulencia, aproximadamente un 70 % de las ejecuciones ya resultan en SKL = 0. Un incremento en el C_n^2 apenas añade unos pocos puntos porcentuales a esta ya elevada tasa de fallos. En consecuencia, el fenómeno atmosférico queda enmascarado y las diferencias de rendimiento entre los escenarios **Soleado**, **Nublado** y **Sin turbulencia** se vuelven prácticamente indistinguibles.

Régimen de fallo sistemático (P_{ec} alto y QBER_i alto):

En el escenario más desfavorable, donde tanto el ruido de fondo como la tasa de error intrínseca son elevados, el ruido interno se convierte en la causa dominante de fallo. Una porción sustancial de las ejecuciones (en torno al 70 %) fracasa antes de considerar el canal atmosférico. La turbulencia adicional solo contribuye marginalmente a la probabilidad de fallo total. Al igual que en el caso anterior, el efecto de la atmósfera queda completamente enmascarado, y las diferencias entre las condiciones de turbulencia son insignificantes.

El análisis demuestra que incrementar cualquiera de los dos parámetros, P_{ec} o QBER_i, degrada el rendimiento y eleva la P_{fallo} . Sin embargo, su impacto combinado es considerablemente más severo y puede llegar a enmascarar por completo los efectos del canal cuántico que se pretende estudiar. Por lo tanto, para maximizar la fiabilidad del enlace y, simultáneamente, preservar la sensibilidad a las condiciones atmosféricas, **es imperativo optimizar ambos umbrales de forma conjunta**.

Conclusiones

Este trabajo presenta una mejora sustancial en la simulación de enlaces de Distribución Cuántica de Claves (QKD) por satélite mediante la optimización del software SatQuMA. Se ha superado una limitación clave del modelo original, que se basaba únicamente en la difracción y la atenuación estática, integrando un módulo de turbulencia atmosférica. La contribución principal es la implementación de un modelo que, a partir de perfiles empíricos de C_n^2 , calcula la varianza de Rytov (σ_R^2) y simula la transmisividad fluctuante del canal (T_{turb}). Esta mejora confiere un realismo significativamente mayor a las simulaciones, permitiendo un análisis riguroso del impacto de la turbulencia en la Tasa de Clave Secreta (SKL).

Gracias a esta nueva capacidad de SatQuMA, el análisis revela que la turbulencia atmosférica degrada el rendimiento del sistema de dos formas interconectadas: reduce la SKL promedio e introduce una alta variabilidad temporal, causada por la prevalencia de desvanecimientos profundos del canal (deep fades).

El análisis de la simulación arroja dos conclusiones fundamentales. En primer lugar, se confirma que la turbulencia atmosférica introduce una atenuación fluctuante en el canal cuántico, lo que provoca una reducción general en la Tasa de Clave Secreta (SKL) promedio en comparación con un escenario estático sin turbulencia.

Más importante aún, el estudio revela que la magnitud de este impacto está fuertemente condicionada por los parámetros de ruido intrínseco del sistema, como la probabilidad de conteo espurio (Pec) y la tasa de error cuántico (QBERi). Se han identificado dos regímenes de operación distintos:

En sistemas de alta fidelidad (PEC y QBER bajos), el efecto de la turbulencia se manifiesta principalmente como un aumento en la variabilidad y en la frecuencia de fallos de generación de clave (SKL=0). En este escenario, el impacto atmosférico es visible en la intermitencia del enlace, pero la SKL obtenida durante los intervalos de buena transmisión se mantiene alta y comparable a la del caso estático.

En sistemas de baja fidelidad (PEC o QBER altos), el sistema ya opera cerca de su umbral de fallo. Aquí, la atenuación adicional inducida por la turbulencia se traduce en una reducción drástica y directa del valor promedio de la SKL. El ruido del sistema se convierte en el factor limitante, y la turbulencia simplemente agrava una tasa de clave ya de por sí degradada, haciendo menos notorio el aumento relativo de fallos (SKL=0) porque el sistema ya fallaba con frecuencia.

A pesar de los avances logrados, este trabajo presenta limitaciones que abren líneas claras para futuras mejoras.

En primer lugar, se identificó una anomalía en la normalización de la transmisividad. El modelo implementado normaliza el valor promedio de la transmisividad fluctuante a la unidad, lo que puede provocar que la transmisividad instantánea supere el valor máximo físicamente posible definido por la atenuación estática (ej. 0.834, según MODTRAN). Este artefacto no físico permite que, en momentos de centelleo positivo ($T_{\text{turb}} > 1$), la SKL simulada con turbulencia supere a la del caso estático. Una corrección futura deberá acotar la transmisividad máxima a valores físicamente realistas para eliminar esta inconsistencia.

En segundo lugar, el modelo actual presenta un carácter determinista en la selección del perfil de turbulencia: para un ángulo de elevación dado, siempre se utiliza el mismo perfil C_n^2 preseleccionado. Para capturar de forma más realista la variabilidad día a día, se propone un enfoque estocástico. Esto implicaría la creación de una base de datos con múltiples perfiles de turbulencia y, para cada simulación, seleccionar uno de ellos aleatoriamente. Este enfoque se desestimó en la fase actual debido a su elevado coste computacional, pero podría implementarse en el futuro mediante la optimización del código o el uso de computación de alto rendimiento (HPC).

El módulo de turbulencia puede adaptarse fácilmente a distintos entornos operativos, incluyendo diferentes tipos de órbitas satelitales (MEO, GEO), diversas ubicaciones de estaciones terrenas y múltiples longitudes de onda. Esto hace de SatQuMA una herramienta flexible y escalable para la planificación y análisis de distintas misiones satelitales de distribución cuántica de claves (QKD).

La validación experimental del modelo, mediante la comparación directa de los resultados simulados con datos reales de campañas de medición satelital, permitirá calibrar y mejorar continuamente su precisión y eficacia.

Finalmente, la versión mejorada del modelo se convierte en un laboratorio virtual idóneo para evaluar integralmente la viabilidad y rendimiento de futuras misiones cuánticas. Esto facilitará la definición de requisitos técnicos específicos (tolerancia al ruido, eficiencia de detección) y la optimización de protocolos cuánticos en escenarios

realistas.