

Guía del Modulador 3D - Trajectory Hub

Índice

1. [Introducción](#)
 2. [Conceptos Fundamentales](#)
 3. [Instalación y Configuración](#)
 4. [Uso Básico](#)
 5. [Presets Disponibles](#)
 6. [Control Avanzado](#)
 7. [Integración con Spat](#)
 8. [Ejemplos Prácticos](#)
 9. [Solución de Problemas](#)
-

Introducción

El **Modulador 3D** es un sistema avanzado que permite controlar la orientación y directividad de las fuentes sonoras en Spat Revolution, independientemente de su posición en el espacio. Esto añade una capa adicional de expresividad y complejidad al movimiento espacial.

¿Qué hace el modulador?

- **Controla la orientación:** Yaw (giro horizontal), Pitch (inclinación) y Roll (rotación)
 - **Modula la apertura:** Controla la directividad del haz sonoro
 - **Es independiente de la posición:** La fuente puede moverse en el espacio mientras su orientación se modula
 - **Permite presets complejos:** Desde respiraciones suaves hasta movimientos caóticos
-

Conceptos Fundamentales

Parámetros P1, P2, P3

El sistema simplifica el control mediante tres parámetros principales:

P1 - Forma de Modulación

Combina:

- **m1**: Forma base (circular, lissajous, espiral, etc.)
- **m2**: Variaciones de escala y traslación

- `m3`: Modos de rotación

P2 - Velocidad (LFO)

- Frecuencia de la modulación en Hz
- Rango típico: 0.1 - 10 Hz

P3 - Directividad (Aperture)

- Apertura del haz sonoro
- Rango: 0.0 (muy direccional) a 1.0 (omnidireccional)

Controladores C1 y C2

- **C1**: Selector/interpolador de presets
 - **C2**: Control de intensidad global (0-100%)
-

Instalación y Configuración

1. Actualizar los archivos del proyecto

```
bash

# Asegúrate de tener la versión más reciente
cd trajectory_hub
git pull

# Reorganizar si es necesario
python tools/reorganize_project.py
```

2. Verificar que el modulador esté habilitado

```
python

# Al crear el motor, asegúrate de habilitar el modulador
engine = EnhancedTrajectoryEngine(
    n_sources=50,
    update_rate=120,
    enable_modulator=True # ¡Importante!
)
```

3. Probar la instalación

```
bash
```

```
# Ejecutar el test de integración  
python test_modulator_integration.py
```

```
# 0 ejecutar con demo en vivo  
python test_modulator_integration.py --demo --demo-duration 60
```

Uso Básico

Ejemplo 1: Aplicar un preset simple

```
python  
  
# Crear un macro  
engine.create_macro("mi_grupo", source_ids=[0,1,2,3,4])  
  
# Aplicar preset de respiración suave  
engine.apply_orientation_preset("mi_grupo", "respiración_suave")
```

Ejemplo 2: Ajustar velocidad e intensidad

```
python  
  
# Cambiar velocidad de modulación (P2)  
engine.set_orientation_lfo("mi_grupo", 0.5) # 0.5 Hz  
  
# Ajustar intensidad global (C2)  
engine.set_orientation_intensity("mi_grupo", 0.7) # 70%
```

Ejemplo 3: Desde el controlador interactivo

```
bash  
  
# En el menú principal:  
20 -> Aplicar Preset de Modulación  
21 -> Ajustar Velocidad (LF0)  
22 -> Ajustar Intensidad
```

Presets Disponibles

1. respiración_suave

- Movimiento circular suave como respiración
- Ideal para: ambientes tranquilos, meditación

- LFO: 0.2 Hz

2. nervioso_aleatorio ⚡

- Vibraciones nerviosas aleatorias
- Ideal para: tensión, suspenso
- LFO: 5.0 Hz

3. espiral_cósmica 🌌

- Espiral expansiva lenta
- Ideal para: espacios amplios, ciencia ficción
- LFO: 0.1 Hz

4. lissajous_complejo ∞

- Figura de Lissajous 3D compleja
- Ideal para: patrones hipnóticos
- LFO: 0.3 Hz

5. péndulo_hipnótico 🕒

- Movimiento pendular hipnótico
- Ideal para: transiciones suaves
- LFO: 0.4 Hz

6. vibración_sísmica 🌍

- Vibraciones sísmicas rápidas
- Ideal para: terremotos, impactos
- LFO: 8.0 Hz

7. flotación_oceánica 🌊

- Movimiento ondulante como en el océano
- Ideal para: escenas submarinas
- LFO: 0.15 Hz

8. rotación_mecánica ⚙️

- Rotación mecánica constante
 - Ideal para: maquinaria, robots
 - LFO: 1.0 Hz
-

Control Avanzado

Interpolación entre Presets

python

```
# Interpolar entre dos presets
# factor: 0.0 = respiración_suave, 1.0 = nervioso_aleatorio
engine.interpolate_orientation_presets(
    "mi_grupo",
    "respiración_suave",
    "nervioso_aleatorio",
    0.5 # 50% de cada uno
)
```

Configuración Manual de Formas

python

```
# Configurar forma directamente
engine.set_orientation_shape(
    "mi_grupo",
    shape="lissajous",
    scale=[1.0, 0.7, 0.5], # Escala en yaw, pitch, roll
    translation=[0.0, 0.0, 0.0] # Offset
)
```

Desfase Temporal entre Fuentes

python

```
# Crear efecto orgánico con desfase
engine.apply_orientation_preset(
    "mi_grupo",
    "espiral_cósmica",
    intensity=1.0,
    time_offset_spread=0.2 # 0.2s entre cada fuente
)
```

Integración con Spat

Parámetros OSC Enviados

El modulador envía estos parámetros a Spat:

```
/source/{id}/yaw      # Giro horizontal (grados)
/source/{id}/pitch    # Inclinación (grados)
/source/{id}/roll     # Rotación (grados)
/source/{id}/aperture # Directividad (0.0-1.0)
```

Verificar la Comunicación

python

```
# Obtener estadísticas OSC
stats = engine.osc_bridge.get_stats()
print(f'Orientaciones enviadas: {stats['parameters_sent']['orientations']}')
```

💡 Ejemplos Prácticos

Ejemplo 1: Bandada de Pájaros Nerviosos

python

```
# Crear bandada
engine.create_macro("pajaros", list(range(20)), formation="sphere")

# Trayectoria errática
engine.set_macro_trajectory("pajaros", "random_walk", speed=0.5)

# Modulación nerviosa
engine.apply_orientation_preset("pajaros", "nervioso_aleatorio",
                                intensity=0.8, time_offset_spread=0.1)
```

Ejemplo 2: Ambiente Submarino

python

```
# Crear grupo de fuentes
engine.create_macro("submarino", list(range(10)), formation="grid")

# Sin movimiento de posición
engine.set_macro_trajectory("submarino", "static")

# Modulación oceánica
engine.apply_orientation_preset("submarino", "flotación_oceánica")
engine.set_orientation_lfo("submarino", 0.1) # Muy lento
```

Ejemplo 3: Máquina Industrial

python

```
# Crear componentes de máquina
engine.create_macro("maquina", list(range(5)), formation="line")

# Rotación mecánica constante
engine.apply_orientation_preset("maquina", "rotación_mecánica")

# Diferentes velocidades para cada componente
for i, sid in enumerate(engine.macros["maquina"].source_ids):
    modulator = engine.orientation_modulators[sid]
    modulator.lfo_frequency = 0.5 + i * 0.2 # Velocidades escalonadas
```

Solución de Problemas

El modulador no funciona

1. Verificar que esté habilitado:

```
python

print(engine.enable_modulator) # Debe ser True
```

2. Verificar que el modulador existe para la fuente:

```
python

sid = 0 # ID de la fuente
if sid in engine.orientation_modulators:
    print("Modulador existe")
    print(f"Activo: {engine.orientation_modulators[sid].enabled}")
```

3. Verificar conexión OSC:

```
python

stats = engine.osc_bridge.get_stats()
if stats['active_targets'] == 0:
    print("⚠ No hay destinos OSC activos")
```

Las orientaciones no se ven en Spat

1. Verificar que Spat esté recibiendo OSC:

- Puerto correcto (normalmente 9000)
- IP correcta (127.0.0.1 para local)

2. Verificar que las fuentes existan en Spat:

- El modulador usa los mismos IDs que las posiciones

- Spat debe tener las fuentes creadas

3. Verificar el umbral de actualización:

```
python

# Reducir umbral si es necesario
engine._orientation_update_threshold = 0.001 # Más sensible
```

Rendimiento bajo con muchas fuentes

1. Reducir la tasa de actualización:

```
python

engine = EnhancedTrajectoryEngine(update_rate=60) # 60 FPS
```

2. Desactivar moduladores no usados:

```
python

engine.toggle_orientation_modulation("mi_grupo", False)
```

3. Usar bundles OSC:

```
python

engine.osc_bridge.targets[0].use_bundles = True
```

Mejores Prácticas

1. **Combinar con moderación:** No todas las fuentes necesitan modulación
2. **Usar desfases:** Crea movimientos más orgánicos
3. **Ajustar intensidad:** Raramente necesitas 100% de intensidad
4. **Pensar en el contexto:** Elige presets apropiados para la escena
5. **Experimentar:** Los mejores efectos vienen de la experimentación

Referencia Rápida

python

Crear macro con modulador

```
engine.create_macro("nombre", [0,1,2], enable_modulator=True)
```

Aplicar preset

```
engine.apply_orientation_preset("nombre", "preset_name")
```

Ajustar velocidad

```
engine.set_orientation_lfo("nombre", 0.5)
```

Ajustar intensidad

```
engine.set_orientation_intensity("nombre", 0.8)
```

Interpolar presets

```
engine.interpolate_orientation_presets("nombre", "preset1", "preset2", 0.5)
```

Toggle on/off

```
engine.toggle_orientation_modulation("nombre", True/False)
```

Obtener presets disponibles

```
presets = engine.get_orientation_presets()
```

¡Disfruta creando movimientos complejos y expresivos con el Modulador 3D! 🎨