

Fehleranalyse

Fehlerhafte PhysX- oder Stage-Einstellungen beim Reset

Problemstellung

Beim Laden einer Szene und dem ersten Reset werden Objekte in der PhysX-Simulation gespawnt. Fehlerhafte Collider, Materialien oder Spawn-Positionen können dazu führen, dass sich Objekte sofort überlappen oder ungültige Zustände erzeugt werden. Dadurch entstehen NaN-Werte, Instabilitäten oder direkte Assertions in PhysX, was die Simulation und oft auch den gesamten Prozess zum Absturz bringt.

Typische Symptome

- PhysX Error: `PxRigidBody::setKinematicTarget: actor must be in a scene`
- Harte Assertion im Log (`physx::shdfnd::checkValid`)
- Crash oder Freeze direkt beim ersten `env.reset()`
- NaN-Werte in Positionen oder Geschwindigkeiten sofort nach Start

Häufige Ursachen

- Interpenetration: Zwei Körper spawnen überlappend, z. B. Apfel bereits im Tisch oder in der Schale.
- Ungültige Collider-Geometrien (z. B. Skalierung auf 0 oder nicht unterstützte Shapes).
- Material-Fehler: Fehlende oder falsch konfigurierte PhysX-Materialien.
- Ungültige Joint-Konfigurationen (z. B. Referenzen auf nicht existierende Bodies).

Minimalbeispiel

```
# Apfel spawnnt exakt in der Schale -> sofortige Interpenetration
spawn("apple", position=(0.0, 0.0, 0.1))
spawn("bowl", position=(0.0, 0.0, 0.1)) # gleicher z-Wert
```

Fix-Patterns

Pattern A: Abstand prüfen

- Spawn-Koordinaten so wählen, dass Objekte mit Mindestabstand erscheinen.
- Plausibilitätsprüfung im Spawn-Skript: Kein Objekt darf innerhalb eines anderen initialisiert werden.

Pattern B: Collider validieren

- Nur unterstützte Geometrien verwenden (Box, Sphere, Convex Mesh).
- Skalierungen ungleich 0 erzwingen.
- Mesh-Collider vorab auf Konvexität prüfen.

Pattern C: Materialien absichern

- Für alle Bodies ein gültiges PhysX-Material registrieren.
- Default-Material als Fallback definieren.

Pattern D: Frühes Abbrechen

- Falls Assets fehlen oder Collider ungültig sind: Sauberer Abbruch mit Fehlerlog, nicht trotzdem starten.

Debug-Playbook

1. Szene mit minimalem Satz an Objekten starten und Schritt für Schritt erweitern.
2. Spawn-Positionen und Collider im Debug-Overlay sichtbar machen.
3. In Python nach dem Reset Objekte inspizieren:

```
for obj in world.objects:  
    print(obj.name, obj.position, obj.collider)
```

4. PhysX-Logs auf NaN-Werte oder Assertions prüfen.
5. Bei hartnäckigen Fehlern Assets isoliert in leerer Stage laden und einzeln testen.

Quick-Checklist

- Keine Überlappung beim Spawn (Mindestabstand erzwingen).
- Collider-Geometrien valide und skaliert.
- PhysX-Materialien gesetzt (oder Default-Material aktiv).
- Früher Check statt Weiterlaufen mit fehlerhaften Assets.
- Nach Änderungen Reset testen, nicht erst Training starten.