



未来への一瞥 AIとMCPによるロボット開発

Fabian Dubois, Artefacts

ROSConJP2025

LET'S GET TO KNOW EACH OTHER



Claude Codeなどでアプリケーション開発ができれば？

- AI駆動の開発環境
- 自然言語からコードへの変換
- インテリジェントなデバッグと最適化
- AIとのリアルタイム協業



う ん う

欠けているもの： フィードバック

LLMはフィードバックなしでは失敗する



AIがコードを生
成

? でも、それが動
作するかどうかやって
知るの？



フィードバック
ループが重要

解決策: シミュレーションテストのMCP

✓ ロボットコードを生成

🔧 シミュレーションテストを実行

📊 パフォーマンスデータを分析

↺ 反復して改善

```
> この結果についておしえてくださいhttps://app.artefacts.com/artefacts/navigation2-ignition-example/job/5e31692f-edf2-46bc-a035-98dd890e4152
```

```
• I'll check the test results for that job to explain what happened.
```

```
• artefacts - get_test_results (MCP)(job_id:  
    "5e31692f-edf2-46bc-a035-98dd890e4152",  
    project:  
    "artefacts/navigation2-ignition-example")  
  
└ [   
    {  
        "scenario_id": "12_follow_waypoints",  
        ... +1394 lines (ctrl+r to expand)    }  
]
```

• このテスト結果は、Navigation2のパラメータ設定を複数のシナリオでテストした結果です。

全体的な結果:

- 45個のシナリオを実行
- 21個が成功 (全4テスト通過)
- 20個が部分的失敗 (test_no_skipped_waypointのみ失敗)
- 4個が重大な失敗 (Nav2の起動自体が失敗)

主要な問題:

1. 重大な失敗 (シナリオ 7, 9, 36, 42):
Nav2が起動できずウェイポイントも追跡できなかった
2. ウェイポイントスキップ: 多くのシナリオでtest_no_skipped_waypointが失敗
(特にcontroller_frequencyが高い設定)

成功パターン:

- controller_frequencyが2-10Hzで良好な結果
- amcl/update_min_aとamcl/update_min_dのバランスが重要
- 最も良い結果: シナリオ11 (amcl/update_min_a=0.2, update_min_d=0.05, controller_frequency=5.0)

推奨設定:

```
controller_frequency: 5-10Hz  
amcl/update_min_a: 0.1-0.2  
amcl/update_min_d: 0.05-0.1
```



Artefacts MCPサーバー 開発中

- AI駆動のロボット開発サイクル
- 継続的なテストと検証
- より高速な反復とデプロイ
- より信頼性の高いロボットシステム

- ログ解析の結果、以下の解決方法が特定されました：

主要な問題と解決策

1. Nav2起動失敗 (シナリオ 7, 9, 36, 42)

原因：AMCLパラメータの組み合わせが位置推定の初期化を妨害

解決策：

```
amcl:
  ros__parameters:
    update_min_a: 0.1      # 0.05は避ける
    update_min_d: 0.15    # 0.05/0.1から増加
    pf_err: 0.03          # 収束性向上のため0.05から削減
```

2. ウェイポイントスキップ問題 (20シナリオ)

原因：ウェイポイント到達検出のタイミング問題

解決策：

```
# follow_waypoints.pyの改善
if feedback and current_wp != feedback.current_waypoint and
feedback.current_waypoint > current_wp:
    print(f"Arrived at {current_wp}'th waypoint @{sim_time}")
    current_wp = feedback.current_waypoint
```

3. ゴール許容値の最適化

```
controller_server:
  ros__parameters:
    general_goal_checker:
      xy_goal_tolerance: 0.15 # 0.25から削減
      yaw_goal_tolerance: 0.15 # 0.25から削減
```

推奨パラメータ組み合わせ

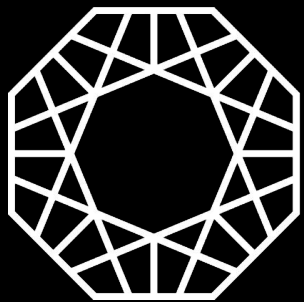
成功率の高い設定：

- update_min_a: 0.1-0.2 + update_min_d: 0.2
- update_min_a: 0.2 + update_min_d: 0.05 (10-15 Hz)

避けるべき組み合わせ：

- update_min_a: 0.05 (全組み合わせ)
- update_min_a: 0.2 + update_min_d: 0.1 (高頻度)

これらの修正により、起動失敗とウェイポイントスキップ問題の両方を解決できます。



Artefacts

ありがとうございました