

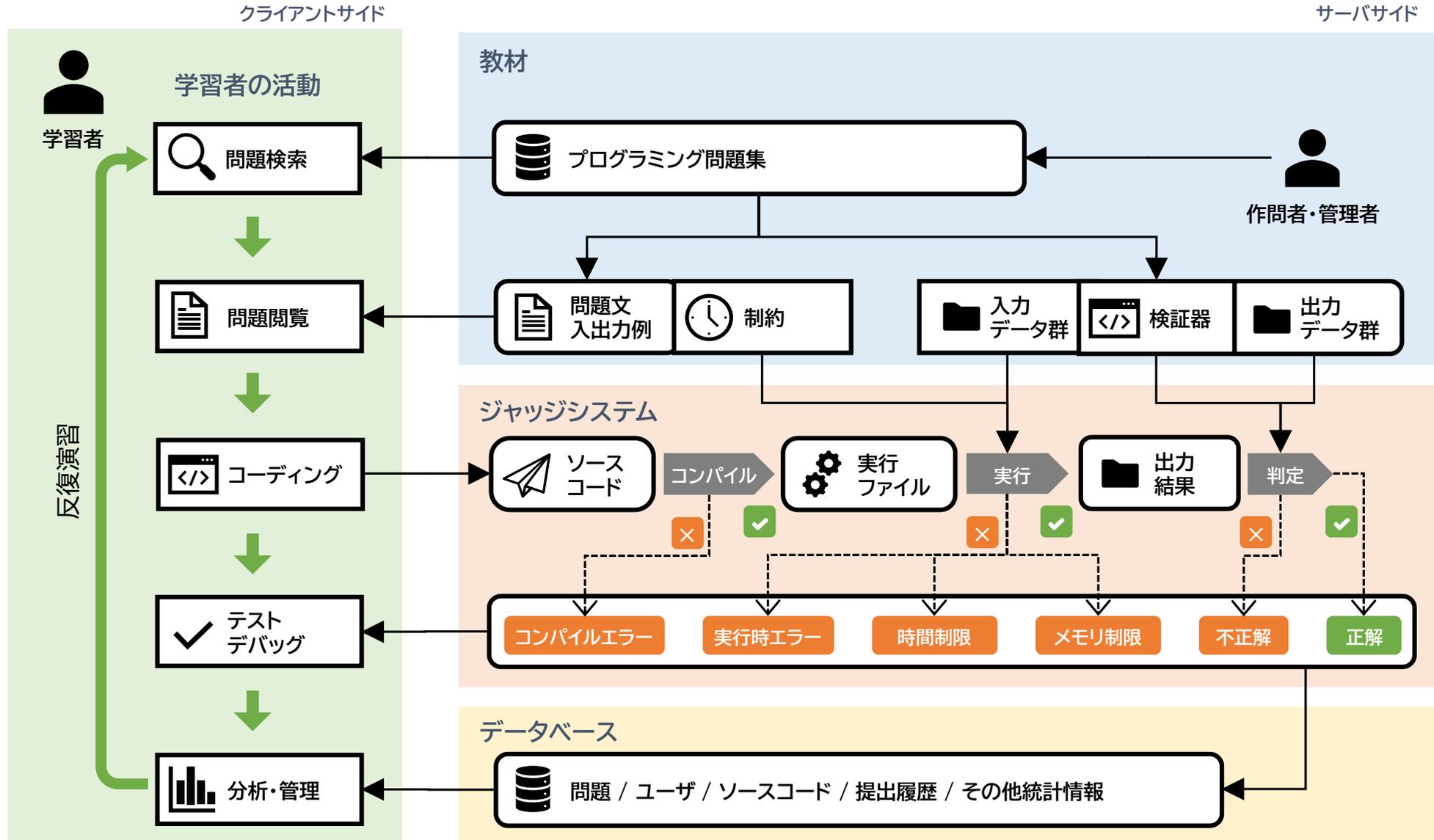
オンラインジャッジシステムによる ロボットソフトウェアの開発・性能評価

第10回会津大学ロボットシンポジウム



渡部有隆
2025年5月30日

オンラインジャッジシステムの仕組み



オンラインジャッジシステムのUI



Aizu Online Judge (AOJ)

<https://onlinejudge.u-aizu.ac.jp/beta/ice>

ワークスペースにて問題閲覧、コーディング、テストを含む全ての工程・活動を管理可能

テストケースによる網羅的な検証

災害対応シミュレーション

3 sec 262144 KB

0469

```
#include<iostream>
#include<cassert>
#include<queue>
#include<vector>
using namespace std;
static const int H_MAX = 300;
static const int W_MAX = 300;
static const int INF = (1<<29);
int H, W, N;
char G[H_MAX+2][W_MAX+2];
int d[H_MAX+2][W_MAX+2];
int c[H_MAX+2][W_MAX+2];

void bfs(int sy, int sx){
    deque<pair<int, int>> Q;
    for (int i = 1; i <= H; i++)
        for (int j = 1; j <= W; j++) d[i][j] = INF;
    d[sy][sx] = 0;
    Q.push_back(make_pair(sy, sx));

    static const int dy[4] = {0, -1, 0, 1};
    static const int dx[4] = {1, 0, -1, 0};

    pair<int, int> u, v;
    while(!Q.empty()){
        u = Q.front(); Q.pop_front();
        int y = u.first;
        int x = u.second;
```

結果: AC

ケース	ステータス	時間	メモリ	入力	出力	ケース名	貼り付け
# 1	AC	0.00 sec	3396 KB	29 B	4 B	00_s...	
# 2	AC	0.00 sec	3148 KB	28 B	4 B	00_s...	
# 3	AC	0.00 sec	3392 KB	26 B	6 B	00_s...	
# 4	AC	0.00 sec	3396 KB	166 B	4 B	01_s...	
# 5	AC	0.00 sec	3388 KB	16 B	4 B	02_c...	
# 6	AC	0.00 sec	3152 KB	16 B	4 B	02_c...	
# 7	AC	0.00 sec	3352 KB	16 B	4 B	02_c...	
# 8	AC	0.00 sec	3160 KB	42 B	4 B	03_g...	
# 9	AC	0.00 sec	3316 KB	46 B	4 B	03_g...	
# 10	AC	0.00 sec	3396 KB	53 B	4 B	03_g...	
# 11	AC	0.00 sec	3336 KB	53 B	4 B	03_g...	
# 12	AC	0.00 sec	3208 KB	85 B	4 B	03_g...	
# 13	AC	0.00 sec	3100 KB	132 B	4 B	03_g...	
# 14	AC	0.00 sec	3460 KB	210 B	4 B	03_g...	
# 15	AC	0.00 sec	3376 KB	46 B	4 B	04_f...	

模範解答の閲覧, 写経演習, コード比較, AIによるレビュー等もサポート

オンラインジャッジシステムの特徴

従来のオンラインジャッジシステム

- 人材育成を目的とした多くの競技会が開催されている
- オンラインのプログラミング学習サービスとして利用可能
- それぞれが学習者のターゲットや目的を持ち、独自の機能と制限のもとに発展



長所

- 最も基本的な標準入出力による採点を行うため、プログラミング言語やプラットフォームに依存しない。
- 初心者から上級者までを想定した幅広いレベルの課題が出題できる。
- テストケースによって網羅的な検証ができる。

短所

- × 正解・不正解及び実行時間等の限られた評価しか提供できない
 - 出題できる課題の形式や分野に限界がある。
- × 学習者側にプログラムの動作の詳細が伝わらない
 - 学習効果改善の観点からも課題が残る。

オンラインジャッジシステムの拡張

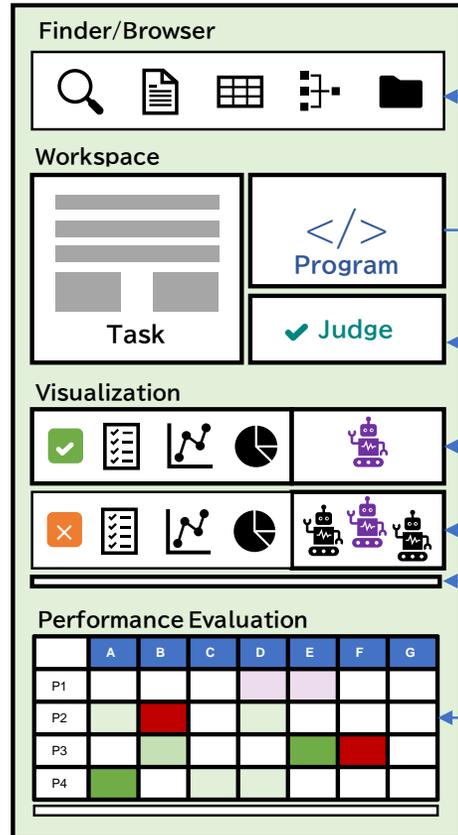
- ロボットソフトウェアの性能評価、開発、教育展開へ向けて

- 評価項目の拡充
- 幾何学的な可視化

オンラインジャッジシステムの拡張

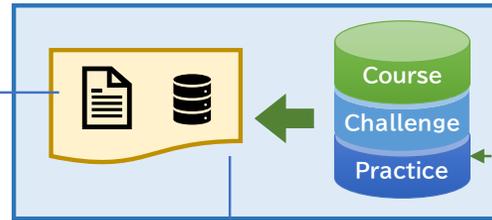
https://www.f-rei.go.jp/assets/contents/20240216_robot.pdf

Smart Programming Environment

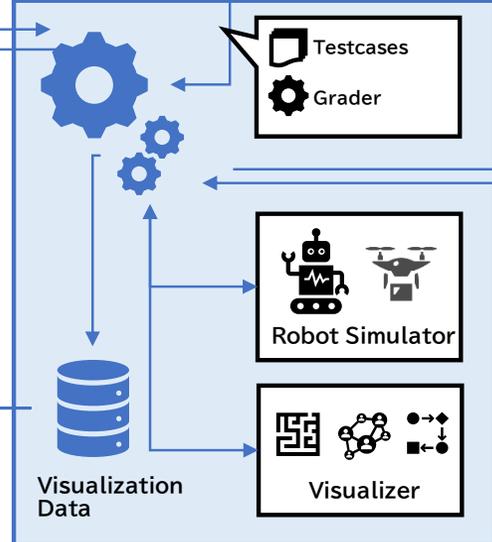


Performance evaluation and visualization of robot software.
ロボットソフトウェアの性能評価と可視化。

Problem Set

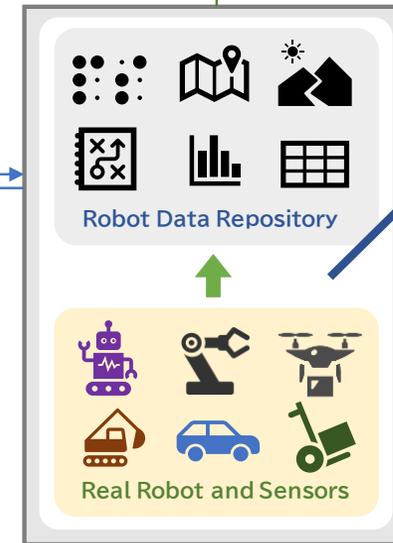


Judge System



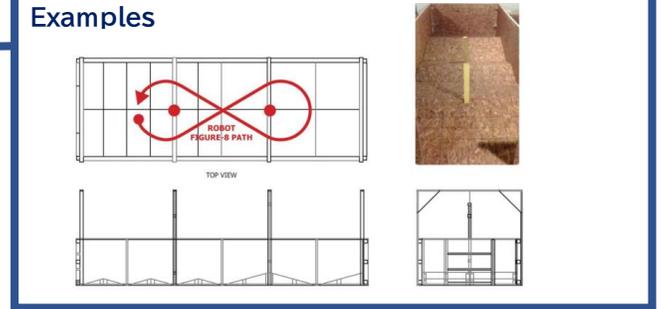
Integration of core technology of Aizu Online Judge (AOJ) and simulator.
Aizu Online Judge (AOJ) のコア技術とシミュレータの統合。

Robots



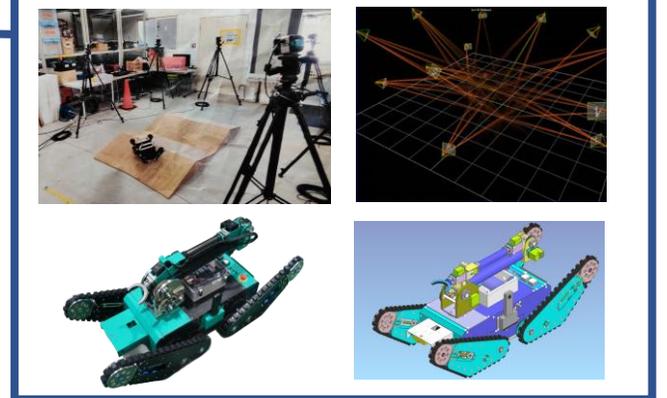
Utilization of data obtained from actual devices and sensors, and the Robot Data Repository (RDR).
実機やセンサーから得られるデータやロボットデータレポジトリ(RDR)の活用。

Examples



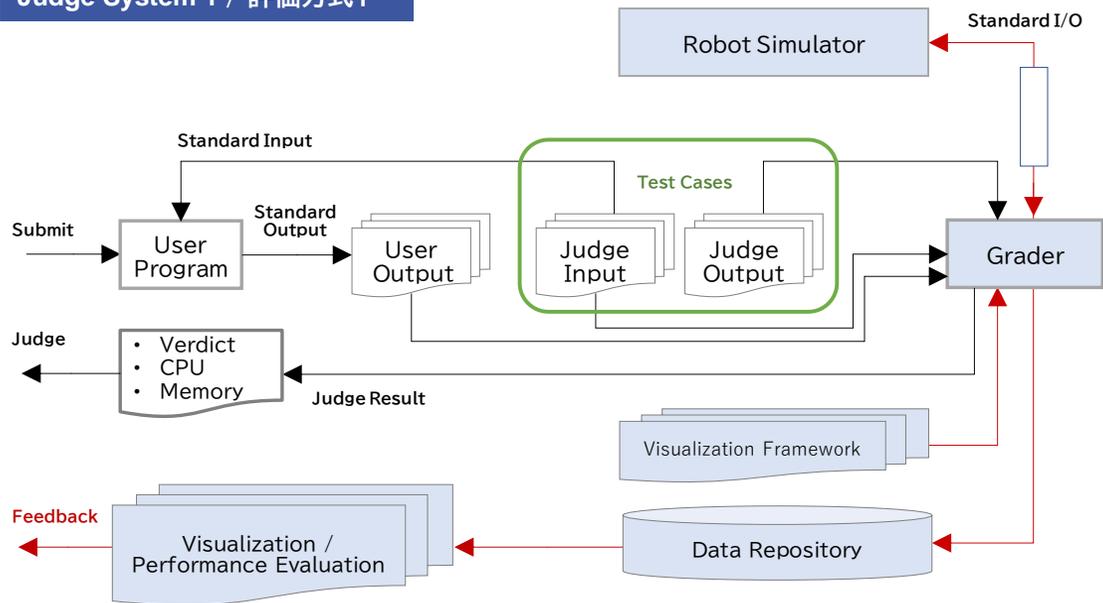
Development and collection of robot programming educational materials.
ロボットプログラミング教材の開発と収録。

Examples



可視化情報・性能評価情報をフィードバック

Judge System 1 / 評価方式1



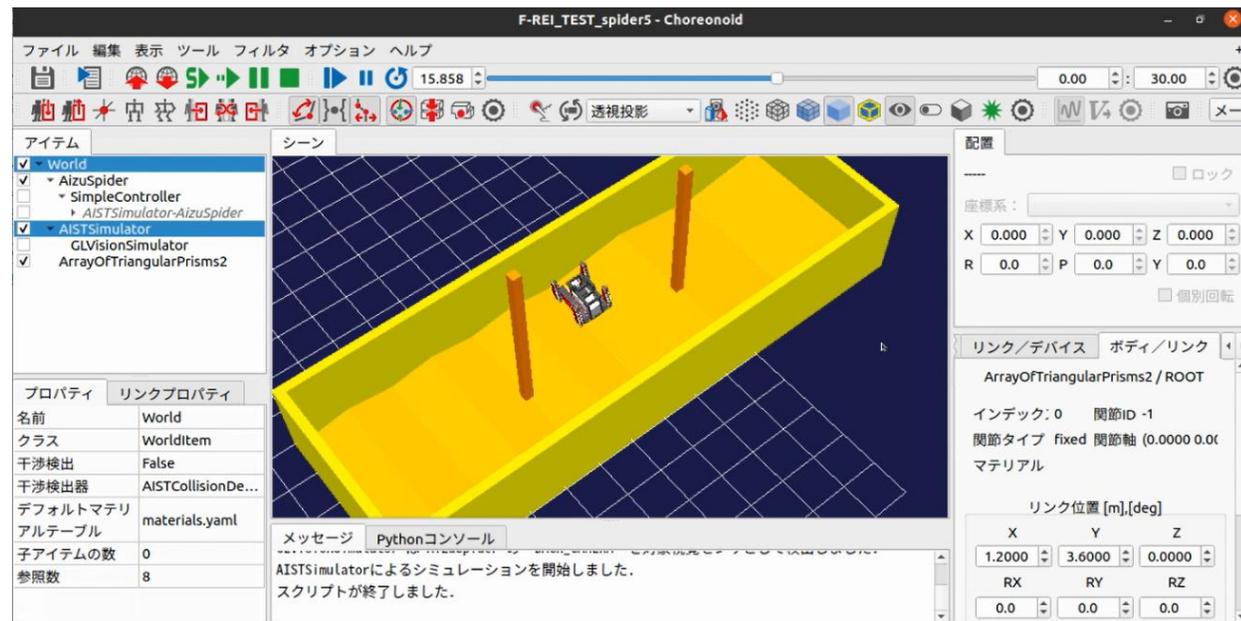
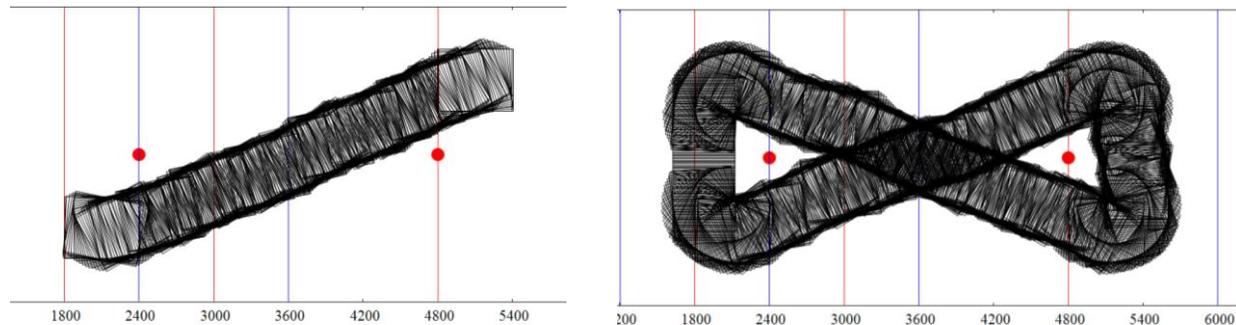
The Grader executes and evaluates user programs using a simulator based on the output of the program and test cases, and then provides feedback in the form of visualization on the execution results and performance evaluation.

Graderがユーザプログラムの出力とテストケースを基にシミュレータによって実行・採点を行い、さらにその実行結果や性能評価に関する可視化情報をフィードバック

プログラム採点システム及びプログラム採点方法

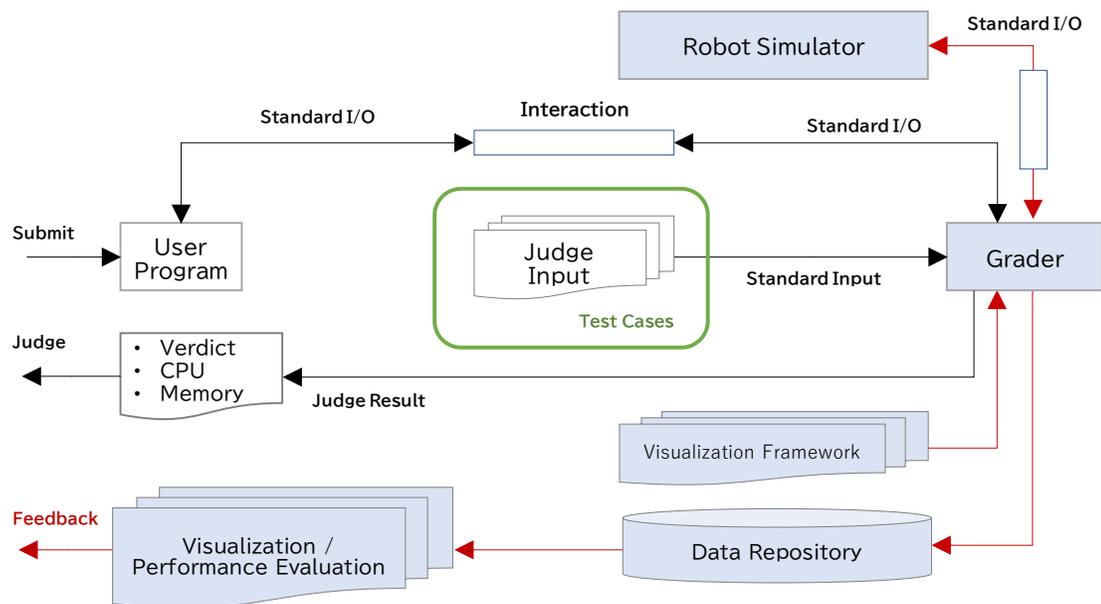
出願人:公立大学法人会津大学
出願番号:特願2023-107217

発明者:渡部 有隆、成瀬 継太郎
出願日:2023/6/29



可視化情報・性能評価情報をフィードバック

Judge System 2 / 評価方式2



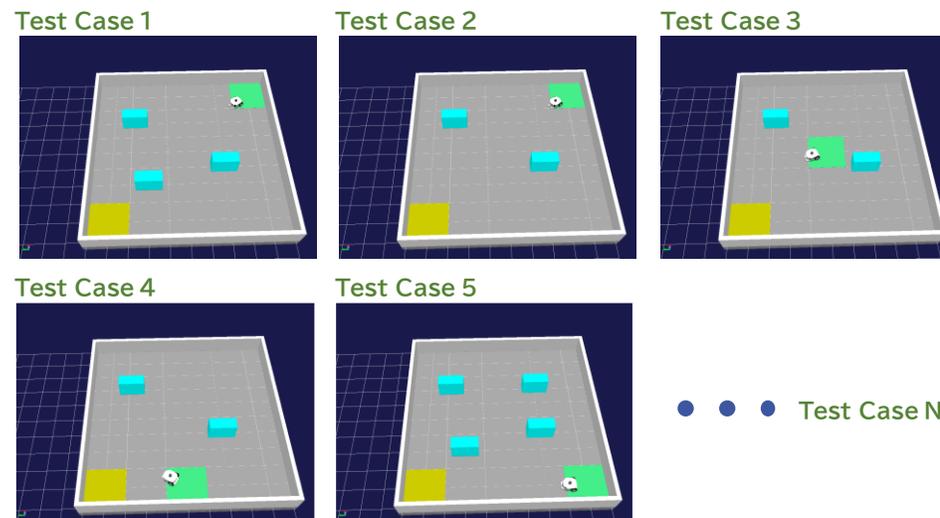
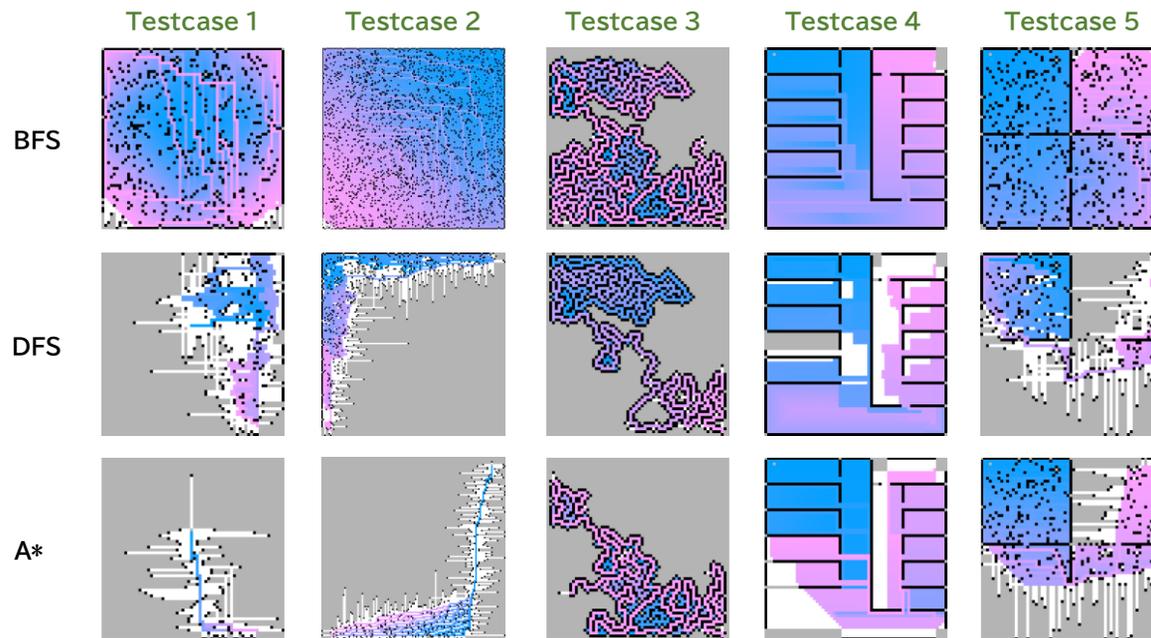
The Grader **executes and evaluates the user program while linking it to the simulator**, and then provides feedback in the form of visualization information about the execution results and performance evaluation.

Graderが**ユーザプログラムとシミュレータを連携させながら実行・評価**を行い、さらにその実行結果や性能評価に関する可視化情報をフィードバック

プログラム採点システム及びプログラム採点方法

出願人: 公立大学法人会津大学
出願番号: 特願2023-107217

発明者: 渡部 有隆、成瀬 継太郎
出願日: 2023/6/29



- 🕒
- 🔗
- 📄
- 💡

問題ID: 8004 時間制限: 10 sec, メモリ制限: 262.144 MB

経路計画

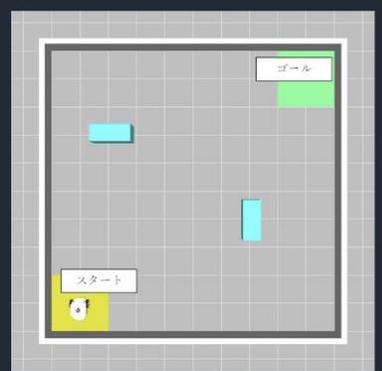
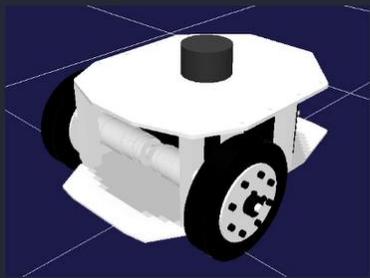
課題

ロボットの形状と運動特性, 作業空間と障害物, ロボットの初期位置と目標位置が与えられる。

また、ロボットには2DLIDARが取り付けられており障害物を認識することができ、センサで認識した障害物との距離が閾値以下となった場合、

1秒後退後旋回を行うものとする。

このとき、ロボットが障害物にあらずに目標位置に到達するような左右の車輪の速度を出力せよ。



ワークスペースのサイズ

作業空間は左下を0,0, 右上をX, -Yとする矩形として与えられる。

スタート地点とゴール地点は作業空間内に1[m]四方の大きさで与えられており、ロボッ

提出履歴 ドキュメント

コーディング in C++

```
1
```

Testcase: Sample Input 1 Sample Input 2

My input	Expected output	Your output
----------	-----------------	-------------

useTest TEST RUN

まとめ：ソフトウェア開発・性能評価・人材育成へ向けて

	シミュレータ	オンラインジャッジシステムの拡張
物理・グラフィックエンジン	○	○ ライブラリ・API利用等でシミュレータの機能を継承
学習コスト	× システム仕様・インタフェースの学習	○ 基礎的なプログラミング技術で利用可能
導入コスト	× ソフトウェアのインストール・環境設定	○ WEBサービスのためインストールや設定が不要
プログラミング言語	△ シミュレータに依存	○ 制限なし(オンラインジャッジで対応している言語)
性能評価	△ テスト・性能評価の習慣化が困難	○ テストケースで網羅的な検証が可能(CI)
他者との性能比較	△ コンペティションを開催する	○ 登録された他のプログラムとの比較が可能
オペレータの評価	○ リアルタイムの操作・モニタリングが可能	× ユーザ画面からの操作不可



オンラインジャッジシステムとロボットシミュレータの統合

- 任意の環境から、プログラミング言語・プラットフォームに依存することなくロボットソフトウェアプログラミングが可能となった
- 様々な評価項目を、網羅的なテストケースによって検証できるようになった
- 性能評価と比較UIによって多角的な評価・分析ができるようになる



ご清聴ありがとうございました。