

# Graph Neural Network Based Action Ranking for Planning :

## Supplementary Material

### 1 Dataset

This supplementary material consists of information on how to access and use the dataset for our paper.

The complete dataset is available at: [https://anonymous.4open.science/r/GABAR\\_dataset-42ED/](https://anonymous.4open.science/r/GABAR_dataset-42ED/)

### 2 Dataset Structure

The dataset is organized into two main directories for training/validation and testing:

Listing 1: Dataset Directory Structure

```
1 dataset/
2 |---- domains/
3 | |---- domain1.pddl/
4 | |---- domain2.pddl/
5 |---- train_val/
6 | |---- domain1_ipcc/
7 | |---- domain2_ipcc/
8 | |---- ...
9 |---- test/
10 | |---- domain1_ipcc_test/
11 | | |---- easy/
12 | | |---- medium/
13 | | |---- hard/
14 | |---- domain2_ipcc_test/
15 | | |---- easy/
16 | | |---- medium/
17 | | |---- hard/
18 |---- ...
```

#### 2.1 Training and Validation Data

The `train_val/` directory contains domain-specific folders, each including PDDL problem files for both training and validation purposes. These files are used for model development and hyperparameter tuning.

#### 2.2 Test Data Organization

The test set comprises:

- **8 domains** across different planning scenarios
- **3 difficulty levels** per domain: Easy, Medium, and Hard
- **100 test files** per difficulty level, totaling 2,400 test instances

## 3 Usage Instructions

This dataset is designed for use with the PDDL Gym framework.

### 3.1 Prerequisites and Installation

- Clone and install PDDL Gym from <https://anonymous.4open.science/r/pddlgyim-0F03/>
- Clone the dataset repository from [https://anonymous.4open.science/r/GABAR\\_dataset-42ED/](https://anonymous.4open.science/r/GABAR_dataset-42ED/)

### 3.2 Setting Up Training Data

For each domain, copy the training and validation files to the corresponding PDDL Gym directory:

Listing 2: Training Data Setup Example

```
1 # Example for gripper domain
2 cp train_val/gripper_ipcc/* pddlgyim/pddlgyim/pddl/gripper_ipcc/
```

### 3.3 Setting Up Test Data

Copy test files based on the desired difficulty level:

Listing 3: Test Data Setup Examples

```
1 # Easy difficulty
2 cp test/gripper_ipcc_test/easy/* pddlgyim/pddlgyim/pddl/gripper_ipcc_test/
3
4 # Medium difficulty
5 cp test/gripper_ipcc_test/medium/* pddlgyim/pddlgyim/pddl/gripper_ipcc_test/
6
7 # Hard difficulty
8 cp test/gripper_ipcc_test/hard/* pddlgyim/pddlgyim/pddl/gripper_ipcc_test/
```

### 3.4 Automated Setup Script

The following bash script automates the dataset setup process:

Listing 4: Automated Setup Script

```
1 #!/bin/bash
2
3 # Set paths
4 DATASET_PATH="path/to/this/dataset"
5 PDDLGYM_PATH="path/to/pddlgyim"
6
7 # Available domains
8 DOMAINS=("blocks" "grripper" "grid" "logistics"
9          "spanner" "visitall" "rovers" "miconic")
10
11
12 # Function to setup a domain
13 setup_domain() {
14     local domain=$1
15     local difficulty=$2
16
17     # echo "Setting up ${domain} domain with ${difficulty} difficulty..."
18
19     # Copy training/validation data
```

```

20 cp -r "${DATASET_PATH}/train_val/${domain}_ipcc/"* \
21     "${PDDLGYM_PATH}/pddlgympddl/${domain}_ipcc/"
22
23 # Copy test data for specified difficulty
24 cp "${DATASET_PATH}/test/${domain}_ipcc_test/${difficulty}/*" \
25     "${PDDLGYM_PATH}/pddlgympddl/${domain}_ipcc/test/"
26 }
27
28 # Copy domain files to correct location:
29 cp -r "${DATASET_PATH}/domains/${domain}_ipcc.pddl" \
30     "${PDDLGYM_PATH}/pddlgympddl/"
31
32 # Usage: setup_domain "gripper" "easy"

```

## 4 Domain Descriptions

The dataset includes eight distinct planning domains:

- **Blocks World** involves manipulating blocks to achieve specific tower configurations. The domain's complexity scales with the number of blocks (6-9 blocks for training/validation, 10-40 blocks for testing).
- **Gripper** requires a robot with two grippers to transport balls between rooms. The domain scales primarily with the number of balls to be moved (5-17 balls for training/validation, up to 100 balls for testing).
- **Miconic** involves controlling an elevator to transport passengers between floors. The domain complexity increases along two dimensions: the number of passengers (1-10 for training/validation, 20-100 for testing) and the number of floors (2-20 for training/validation, 11-30 for testing).
- **Logistics** involves transporting packages between locations using trucks (for intra-city transport) and airplanes (for inter-city transport). The domain scales with both the number of cities (4-8 for training/validation, 15-30 for testing) and packages (3-9 for training/validation, 10-24 for testing).
- **Visitall** requires an agent to visit all cells in a grid. The domain scales with grid size (9-49 cells for training/validation, up to 400 cells for testing - testing problems 8 times larger than the training dataset).
- **Grid** involves navigating through a grid where certain doors are locked and require specific keys to open. The number of locks (3) and keys (5) remain the same across training and testing while varying the size of the grid (7×9 for training/validation to 11×14 for testing - a 150% increase in cells).
- **Spanner** requires tightening nuts at one end of a corridor with spanners collected along the way. Number of locations varies from 10-20 in training and testing, the number of spanners varies from 2-10 in training and 11 to 40 in testing.
- **Rovers** requires multiple rovers equipped with different capabilities to perform experiments and send results back to the lander. Training and validation instances use 2-5 rovers and 3-10 waypoints; those for testing have 3-7 rovers and 11-70 waypoints.

## 5 Evaluation Protocol

For consistent evaluation across different methods:

1. **Training Phase:** Utilize files from `train_val/{domain}_ipcc/` for model training
2. **Testing Phase:** Evaluate performance across all three difficulty levels:

- *Easy*: Fundamental problems with minimal complexity
- *Medium*: Intermediate complexity requiring strategic planning
- *Hard*: Complex scenarios with significant planning challenges

## 6 File Format Specifications

All files adhere to standard PDDL (Planning Domain Definition Language) format:

- **Domain files**: Define predicates, actions, and domain-specific constraints
- **Problem files**: Specify initial states and goal conditions for planning instances

## 7 Technical Notes

- The dataset maintains consistency with PDDLGym’s expected directory structure
- Each test difficulty level provides exactly 100 instances for statistical significance
- Domain naming follows the convention `{domain_name}_ipcc` for training data and `{domain_name}_ipcc_test` for test data

## 8 All Code and dataset

- All code is available at : <https://anonymous.4open.science/r/ltp-4741/README.md> along with installation instructions
- All data is available at : [https://anonymous.4open.science/r/GABAR\\_dataset-42ED](https://anonymous.4open.science/r/GABAR_dataset-42ED) along with usage instructions