### A. PREPARING THE ENVIRONMENT

Before running the experiments, we must prepare the simulation environment. This begins by downloading the framework from its GitHub repository[1]. Next, we must define two paths for the experiments: one for the framework executable and one for saving the data. In the first case, the file is located at:

```
(download folder\)RAWSimO.CLI\bin\Debug\net6.0\
    ↪ RAWSimO.CLI.exe
```

The second path can be any location, but keep in mind that the simulation files have long names, so one should favor locations close to the disk root. *RAWSim-O* has the ability to run a single simulation at once. This can be done with the following command:

```
(local path\)RAWSimO.CLI.exe <.xinst> <.xsett>
    ↪ <.conf> <StatisticsDir> <Seed>
```

Using instance 11 as an example, and assuming a simulation window of 24 hours, with the `Random` configuration for both subproblems, we have that:

- Instance (.xinst): `1-1-1-1-3094.xinst`
- Setting (.xsett): `SimpleItem-Fill-200-200-86400.xsett`
- Controller (.xconf): `Random_PPFAR-TABalanced-SAActivateAll-ISRandom-PSRandom-RPDummy-OBPodMatching-RBSamePod-MMNoChange.xconf`
- StatisticsDir: Folder for saving the results, *e.g.*, `testOne`.
- Seed: For the random number generator, *e.g.*, `1`.

This results in:

```
(local path\)RAWSimO.CLI.exe 1-1-1-1-3094.xinst
    ↪ SimpleItem-Fill-200-200-86400.xsett
    ↪ Random_PPFAR-TABalanced-SAActivateAll-
    ↪ ISEmptiest-PSRandom-RPDummy-OBPodMatching-
    ↪ RBSamePod-MMNoChange.xconf testOne 1
```

Since we run a lot of simulations, we generated batch files (using Matlab) that indicate all the simulations for a given set. For example, Set 1 contains instance 11. So, the batch file for running the experiments in Set 1 is `BatchFile_Main_Stage_Set_1_inst11_12_15_30_seed1_2_3_7_24h.bat`. This file can be generated by adapting the Matlab file `namebatch_itemStorage.m`. Considering that there are several modifications that must be applied, and for the sake of brevity, we omit a detailed description of each change. In turn, we invite the interested reader to check the aforementioned Matlab file, which includes comments explaining the changes that must be incorporated.

### B. PROCESSING THE DATA

Since each simulation outputs several files, we also generated batch files to process the output data and extract the files that contain performance statistics. This is done in two steps. First, the script `namebatch_itemStorage.m` generates a batch file that unifies the statistics-related file from each experiment in a single folder. Note that this is the same file as in the previous section. Once again, we note that the Matlab file has been commented to clearly indicate where each task begins and ends. Second, the script `Extract_Data_experiments_itemStorage_set1.m` process such files and creates the final tables with the performance data for each configuration and technique.

### C. INSTANCES CONSIDERED IN THIS WORK

The instances were initially separated into 4 groups (Table 1): Set 1 holds instances 11, 12, and 15 through 30; Set 2 has instances 31-40; Set 3 has instances 41-56; Set 4 has instances 57-72. Each set represents different kinds of warehouse configuration. The first set captures extreme cases, *e.g.*, a warehouse with a single workstation and more than 3000 pods. The second set focuses on cases with few stations (1, 6, or 12), and with a significantly reduced number of pods and robots. The third set aims to replicate smaller warehouses, where the number of stations is balanced, but the number of pods and robots is small. The last set increases the number of pods (up to 673) while preserving the balance between stations. These scenarios were designed based on the warehouse size classifications (small, medium, and large) proposed by [1].

Our simulations consider 24 hours of warehouse operation. The real simulation time differs from machine to machine. Depending on their specifications, one simulation may run faster, but the results will be the same as long as both use the same configuration. We made sure this was true so that we could run these batch files on different machines. Since we consider a high volume of experiments, it took several days to complete them all.

**TABLE 1. Properties of each set of instances considered in this work.**

| Set | Instances | Workstations | Bots [min, max] | Pods [min, max] |
|---|---|---|---|---|
| 1 | [11,12, 15-30] | [1, 6, 12] | [1, 32] | [142, 3094] |
| 2 | [31-40] | [1, 6, 12] | [1, 12] | [618, 1190] |
| 3 | [41-56] | [2, 4] | [3, 6] | [102, 673] |
| 4 | [57-72] | [2, 4] | [10, 75] | [102, 673] |

## REFERENCES

[1] M. Merschformann, L. Xie, and D. Erdmann, "Path planning for robotic mobile fulfillment systems," *ArXiv*, vol. abs/1706.09347, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:25541161

[1] https://github.com/merschformann/RAWSim-O