PRISM Documentation

Release 0.0.0.0

John Harwell

Oct 08, 2021

CONTENTS:

1 XML Configuration **1**

1.1 Controller XML Configuration 1

1.2 Loop Functions XML Configuration 2

1.3 XML Conventions 6

2 Other Projects (in descending probability of interest) **7**

CHAPTER

ONE

XML CONFIGURATION

1.1 Controller XML Configuration

The following controllers are available:

- – fcrw_bst
  - – construction
  - – CRW Foraging + single target building.

The following root XML tags are defined under <params>.

1.1.1 lane_alloc

- Required child attributes if present: policy.
- Required child tags if present: none.
- Optional child attributes: [interference_window].
- Optional child tags: none.

XML configuration:

```
<lane_alloc
    policy="random|lru|closest|min_interference">
    interference_window="INTEGER"
/>
```

policy - The lane allocation policy to use. Valid values are:

- random - Choose a random lane each time.
- lru - Choose the least recently visited lane each time by allocating lanes in a round robin fashion. Initialized to a random lane for each robot to prevent crowding at the start of simulation.
- closest - Choose the lane closest to the robot's current location when the allocation algorithm is run.
- min_interference - Choose the lane which the robot has experienced the minimum average interference (i.e., having to wait for other robots) while in the lane. The average interference is calculated using a sliding window of interference_window timesteps. If multiple lanes have the same average, a random one is selected.

1.2 Loop Functions XML Configuration

The following root XML tags are defined under <loop_functions> in addition to the ones specified in COSM.

Root XML tag	Mandatory For?	Description
output	All	See COSM docs.
arena_map	All	See COSM docs.
temporal_variance	None	See COSM docs.
visualization	None	See COSM docs.
oracle_manager	None	See COSM docs.
structure3D_builder	All controllers	Parameters for the structure builder.
construct_targets	All controllers	Parameters defining the shape/size of the structures to be built.

Any of the following attributes can be added under the metrics tag in place of one of the <append>,<create>, <truncate> tags. Not defining them disables metric collection of the given type.

XML attribute	Description
fsm_collision_counts	See COSM docs.
fsm_collision_locs2D	See COSM docs.
fsm_movement	See COSM docs.
block_acq_counts	See COSM docs.
block_acq_locs	See COSM docs.
block_acq_explore_locs	See COSM docs.
block_acq_vector_locs	See COSM docs.
block_transport	See COSM docs.
task_distribution	See COSM docs.
swarm_dist_pos2D	See COSM docs.
swarm_dist_pos3D	See COSM docs.
tv_population	See COSM docs.
block_manipulation	Free block pickup/block placement counts/penalties.
structure_progress	Counts of block placement for the within the structure.
structure_state	3D occupancy map of cells within the structure with block placement status.
structure_subtargets	Counts of block placement/progress for all subtargets within the structure.

1.2.1 structure3D_builder

- Required by: all.
- Required child attributes if present: [build_src].
- Required child tags if present: none.
- Optional child attributes: [static_build_interval, static_build_interval_count].
- Optional child tags: [usage_penalty].

XML configuration:

```
<structure3D_builder
    build_src="loop|robot"
    static_bulid_interval="INT"
    static_build_interval_count="INT"
/>
```

- `build_src` - The source of block placements for the structure. Valid values are:
  - `loop` - The structure will be built statically each timestep by the loop functions without robot involvement. Mainly intended as a debugging tool for initial bring up and structure invariant testing.
  - `robot` - The structure will be built by robots.
- `static_build_interval` - How many timesteps between invocations of the static builder in the loop functions. Only used if `build_src` is `loop`. Defaults to 1 if omitted.
- `static_build_count` - How blocks to place on the structure when the builder is invoked at the start of an interval. Only used if `build_src` is `loop`. Defaults to 1 if omitted.

1.2.2 construct_targets

- Required by: all.
- Required child attributes if present: none.
- Required child tags if present: none.
- Optional child attributes: none.
- Optional child tags: [`ramp`, `rectprism`].

XML configuration:

```
<construct_targets>
    <ramp>
        ...
    </ramp>
    <rectprism>
        ...
    </rectprism>
    ...
</construct_targets>
```

- `ramp` - Defines a ramp subtarget.
- `rectprism` - Defines a rectangular prism subtarget.

construct_targets/ramp

- Required by: [none].
- Required child attributes if present: [`anchor`, `grid`, `id`, `orientation`].
- Required child tags if present: [`ramp_blocks`, `cube_blocks`].
- Optional child attributes: none.
- Optional child tags: none.

XML configuration:

```
<construct_targets>
    ...
    <ramp anchor="FLOAT,FLOAT,FLOAT"
          id="ramp0"
```

(continues on next page)

(continued from previous page)

```
          orientation="FLOAT">
        <grid>
            ...
        </grid>
        <ramp_blocks>
            ...
        </ramp_blocks>
        <cube_blocks>
            ...
        </cube_blocks>
    </ramp>
    ...
</construct_targets>
```

- `anchor` - X,Y,Z coordinates of the lower left hand corner of the structure specifying its absolute location in the arena.
- `id` - A UUID for the structure.
- `orientation` - The angle in radians between the X axis of the structure and the X axis of the arena. Can be 0 or pi/2; other values will cause an error.

`construct_targets/ramp/grid`

- Required by: all.
- Required child attributes if present: [`resolution`, `size`].
- Required child tags if present: none.
- Optional child attributes: none.
- Optional child tags: none.

XML configuration:

```
<ramp>
    ...
    <grid
        resolution="FLOAT"
        size="X, Y, Z"/>
    ...
</ramp>
```

- `resolution` - The resolution that the structure will be represented at, in terms of the size of grid cells. Must be the same as the value passed to the robot controllers.
- `size` - The size of the bounding box containing the ramp structure.

construct_targets/ramp/ramp_blocks

- Required by: [none].
- Required child attributes if present: none.
- Required child tags if present: none.
- Optional child attributes: none.
- Optional child tags: [`ramp_block`].

XML configuration:

```
<ramp>
    ...
    <ramp_blocks>
        <ramp_block cell="INT,INT,INT" id="0">
        <ramp_block cell="INT,INT,INT" id="1">
        ...
    </ramp_blocks>
    ...
</ramp>
```

Defines the ramp blocks needed to construct the specified ramp construction target. The X,Y,Z coordinates for each `cell` cell attribute are RELATIVE to the structure `anchor` (0,0,0 by convention).

construct_targets/ramp/cube_blocks

- Required by: [none].
- Required child attributes if present: none.
- Required child tags if present: none.
- Optional child attributes: none.
- Optional child tags: [`cube_block`].

XML configuration:

```
<ramp>
    ...
    <cube_blocks>
        <cube_block cell="INT,INT,INT" id="0">
        <cube_block cell="INT,INT,INT" id="1">
        ...
    </cube_blocks>
    ...
</ramp>
```

Defines the cube blocks needed to construct the specified ramp construction target. The X,Y,Z coordinates for each cube block `cell` attribute are RELATIVE to the structure `anchor` (0,0,0 by convention).

construct_targets/cube/cube_blocks

Same as for `construct_targets/ramp/cube_blocks`.

1.3 XML Conventions

- Multiple choices for an XML attribute value are separated by a `|` in the example XML.
- XML attributes that should be floating point are specified as `FLOAT` in the example XML (acceptable range, if applicable, is documented for each individual attribute).

CHAPTER

TWO

OTHER PROJECTS (IN DESCENDING PROBABILITY OF INTEREST)

- FORDYCA
- SIERRA
- COSM
- RCPPSW
- RCSW