Data structure for World Model sharing in MSL

1. Introduction

The need for ever growing data standardization among the MSL community is an important step towards the ability of extracting practical and real data for offline evaluation of own and opponent team performance during tournaments and to provide an effective way of allowing easy ad-hoc mixed teams formation in the future.

With this in mind, it has been agreed upon, during the 2013 MSL Workshop held in Kassel, Germany, to start by establishing a first data structure, to be used as a standard by every MSL team, to allow useful data logging during the games. This data may in the future be combined with a ground truth system, whenever a reliable one is made available to the MSL competitions.

It was common agreed that this approach is feasible in a short term and may already provide the teams with very useful information towards research evolution in the league.

2. Data Type and coordinate system definitions

All data regarding robots, obstacles or ball are to be defined in a world three dimensional Cartesian coordinate system defined according to the following rules (see figure 1):

- for any team, the coordinate system is defined as viewed from the own side of the field;
- the coordinate system follows the right system principle;
- the *xy* plane is coincident with the field floor;
- the origin of the coordinate system is coincident with the center of the field;
- the *x* axis is coincident with the center of the mid-field line and positive to the right;
- the *y* axis is positive towards the opponent goal;
- the z axis is positive above the ground;
- angles are measured from the *x* axis, or from any other line parallel to the *x* axis, in the counter clockwise direction and are defined in the interval $[0 .. 2\pi[$;
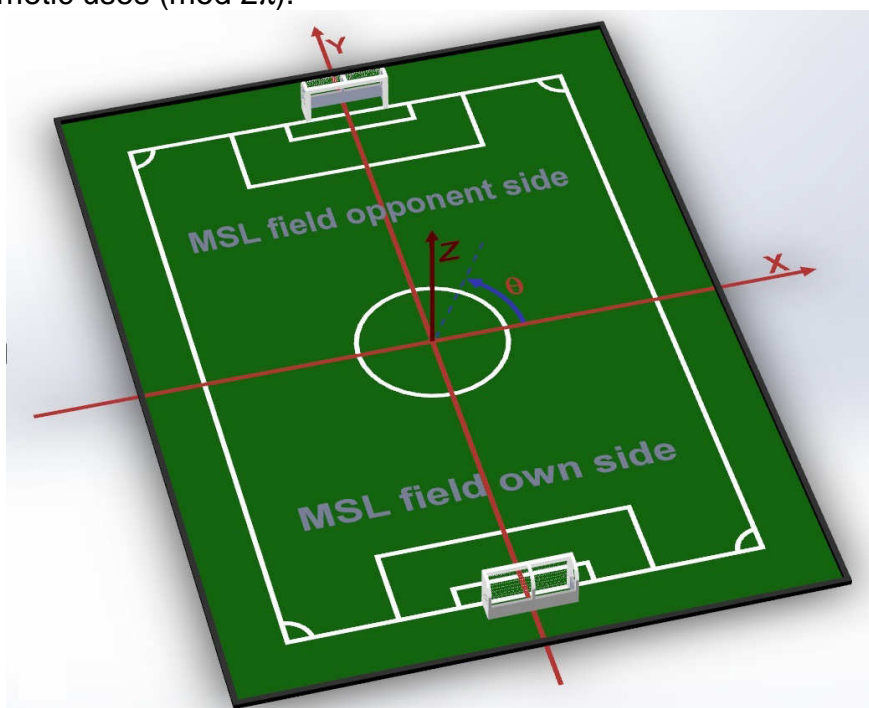- angle arithmetic uses (mod 2π).

Fig. 1 - Three dimensional Cartesian coordinate system for world coordinate representation

Since, for any team, the coordinate system is defined as viewed from the own side of the field, to obtain any coordinates from the opponent team data structure a transformation must be performed.

This transformation can be expressed as

$$[x' \quad y' \quad z'] = [x \quad y \quad z] \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad \theta' = \theta + \pi \pmod{2\pi}$$

Besides the above rules the following ones also apply:

- positions are always defined as a triplet [x, y, z] in the above defined three dimensional Cartesian coordinate system;
- positions are defined in meters, with a minimum of three decimal digits of resolution;
- angles are defined in radians, with a minimum of four decimal digits of resolution;
- velocities are defined in *m/s* as a vector starting at the coordinate system origin and defined by its cartesian representation $\mathbf{v_i} = [x_i, y_i, z_i]$.
- velocity of a given object (e.g. the ball) is obtained by the translation of origin of the aforementioned vector to the object position coordinates at a given instant.
- robot heading is represented as an angle between the line that connects the center of the robot projection on the floor and the front of the robot and the X axis (see figure 2).
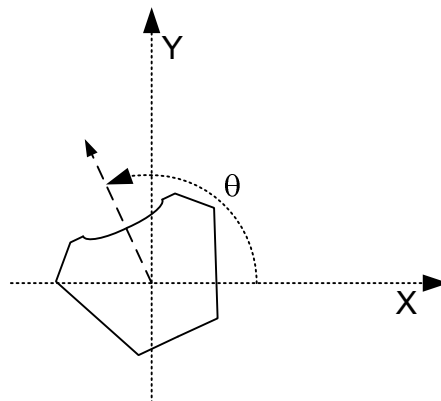

Fig 2 – Robot heading measured in the *xy* plane translated to the center of the robot

3. Information to be provided

Upon every refbox request (which includes a time stamp), each base station is required to answer with a data structure containing the following information:

- Time Stamp[1] (received from the refbox);
- Team name;
- Current ball position in the coordinate system;
- Current ball velocity vector;
- For each of the team five robots:
  - Robot number;

_____
[1] Time stamp is the same value received from the refbox

- o Robot role[2];
- o Current robot position and heading in the coordinate system;
- o Current robot velocity vector[3];
- For each of the opponent team five robots:
  - o Robot number;
  - o Estimated current robot position and heading in the coordinate system;
  - o Estimated current robot velocity vector;

Apart from the two first fields, which are mandatory, all other fields may be not defined or partially not defined depending on the ability of the team to determine them. The rules for constructing this structure, including missing data, is defined in the next section.

If the refbox fails to receive the information from one or both of the base stations during the sampling period, it will fill in an empty data structure and store it in the logging file.

The sampling period will be 100ms (10 requests per second).

The log file will be provided to both teams at the end of each game.

4. Data structure formal definition

The data structure provided by each base station upon request from the refbox will follow the JSON (JavaScript Object Notation) notation. Detailed information about the grammar of this lightweight data-interchange format may be found in http://www.json.org/.

All information that composes the data structure (which will be an object in JSON) is text based, and its embedded structuring using simple markers such as "{" and "}" results in a light and easy to parse and/or read data-interchange packet. A large set of off-the-shelf tools for building or parsing JSON structures is also readily available for a large set of different programming languages.

The format of the MSL-JSON object instance to be delivered by each base station will have the structure depicted in the following pages:

---

[2] String used to identify the role of the player: GoalKeeper, MidFielder or Out (when player is out of the field)

[3] Robot velocity vector is the desired velocity (resulting from the set points applied to the wheels), not the estimated velocity.

```
{
    "name" : "MSLWMD",
    "description":"MSL World Model data",
    "type": "object",
    "properties":{
        "Team Name":{"type":"string",
                        "description":"Name of the team"
        },
        "Time Stamp":{"type":"integer",
                        "description":"same value received in the request"
        },
        "Ball":{
            "type":"object",
            "properties":{
                "Position":   {"type": "array",
                    "items": {
                        "type":["number", "null"],
                            "description":"Ball x position",
                        "type":["number", "null"],
                            "description":"Ball y position",
                        "type":["number", "null"],
                            "description":"Ball z position"
                    }
                },
                "Velocity":{"type": "array",
                    "items": {
                        "type":["number", "null"],
                            "description":"Ball x velocity component",
                        "type":["number", "null"],
                            "description":"Ball y velocity component",
                        "type":["number", "null"],
                            "description":"Ball z velocity component"
                    }
                }
            }
        }
        "My Team":{
            "type": "array",
            "items": {
                "type":"object",
                    "properties":{
                        "id": { "type": "number",
                            "description": "robot number from 1 to 5"
                        },
                        "Role": {"type": ["string", null],
                            "description": ""GoalKeeper"", ""MidFielder"" or ""Out"""
                        },
                        "Position":   {"type": "array",
                            "items": {
                                "type":["number", "null"],
                                    "description":"Own robot x position",
                                "type":["number", "null"],
                                    "description":"Own robot y position",
                                "type":["number", "null"],
                                    "description":"Own robot z position",
                                "type":["number", "null"],
                                    "description":"Own robot heading"
                            }
                        },
                        "Velocity":{"type": "array",
                            "items": {
                                "type":["number", "null"],
                                    "description":"own robot x velocity component",
                                "type":["number", "null"],
                                    "description":"own robot y velocity component",
                                type":["number", "null"],
                                    "description":"own robot z velocity component"
                            }
                        }
                    }
                }
            }
```

```
            "minItems": 5
        }
    "Other Team":{
        "type": "array",
        "items": {
            "type":"object",
            "properties":{
                "id": { "type": "number",
                    "description": "robot number from 1 to 5"
                },
                "Position":   {"type": "array",
                    "items": {
                        "type":["number", "null"],
                            "description":"Opp robot x position",
                        "type":["number", "null"],
                            "description":"Opp robot y position",
                        "type":["number", "null"],
                            "description":"Opp robot z position",
                        "type":["number", "null"],
                            "description":"Opp robot heading"
                    }
                },
                "Velocity":{"type": "array",
                    "items": {
                        "type":["number", "null"],
                            "description":"opp robot x velocity component",
                        "type":["number", "null"],
                            "description":"opp robot y velocity component",
                        "type":["number", "null"],
                            "description":"opp robot z velocity component"
                    }
                }
            }
        },
        "minItems": 5
    }
}
    "required": [all"]
}
}
```

An example of the JSON schema for the MSL World Model Data, according to the above description, can be seen in the following page:

```
{
    "Team Name": "CAMBADA",
    "Time Stamp": 12367,
    "Ball": {
            "Position":  [3.671, -2.100, 0.000],
            "Velocity":  [null, null, null]
          },
    "Own Team":  [
                    {
                      "id": 1,
                      "Role": "GoalKeeper",
                      "Position":  [0.252, -8.800, 0.000, 1.5803],
                      "Velocity":  [0.000, 0.000, 0.000]
                    }, {
                      "id": 2,
                      "Role": "MidFielder",
                      "Position":  [0.252, -8.800, 0.000, 1.5803],
                      "Velocity":  [0.000, 0.000, 0.000]
                    }, {
                      "id": 3,
                      "Role": "MidFielder",
                      "Position":  [-2.476, 3.600, 0.000, 1.7512],
                      "Velocity":  [-0.834, 1.530, 0.000]
                    }, {
                      "id": 4,
                      "Role": "Out",
                      "Position":  [null, null, null, null],
                      "Velocity":  [null, null, null]
                    }, {
                      "id": 5,
                      "Role": "MidFielder",
                      "Position":  [0.252, -8.800, 0.000, 1.5803],
                      "Velocity":  [0.000, 0.000, 0.000]
                    }
                 ],
    "Other Team":[
                    {
                      "id": 1,
                      "Position":  [-0.055, 8.970, 0.000, null],
                      "Velocity":  [null, null, null]
                    }, {
                      "id": 2,
                      "Position":  [null, null, null, null],
                      "Velocity":  [null, null, null]
                    }, {
                      "id": 3,
                      "Position":  [null, null, null, null],
                      "Velocity":  [null, null, null]
                    }, {
                      "id": 4,
                      "Position":  [null, null, null, null],
                      "Velocity":  [null, null, null]
                    }, {
                      "id": 5,
                      "Position":  [null, null, null, null],
                      "Velocity":  [null, null, null]
                    }
                 ]
}
```