Project Number: XX

Start Date of Project: 2021/01/01

Duration: 18 months

**Type of document D2.1 – V1.0**

**Project Management Guidelines**

|  |  |
| --- | --- |
| Dissemination level | XXXXX |
| Submission Date | 2021-12-31 |
| Work Package | WP2 |
| Task | TX:X |
| Type | XXXX |
| Version | 1.0 |
| Author | XXXXXX |
| Approved by | XXXXXX |

**Executive Summary**

**Table of Content**

[1 Section 5](#_heading=h.gjdgxs)

[1.1 Subsection 5](#_heading=h.30j0zll)

[1.1.1 Subsubsection 5](#_heading=h.1fob9te)

**Abbreviations and Acronyms**

|  |  |
| --- | --- |
| AMGA | Annotated Model Grant Agreement |
| BEN | Beneficiary |
| CA | Consortium Agreement |
| CO | Coordinator |
| DoA | Description of Action |
| EC | European Commission |
| PR | Periodic Report |
| SyGMa | System for Grant Management |
| TS | Technical Staff |
| WP | Work Package |
| XXXXX | XXXXX |
| XXXXX | XXXXX |

# Environment Modelling

When modelling the environment, we initially consider a multi-room layout as depicted in Figure 1. This environment represents a relatively simple environment consisting of an open plan living, kitchen and dining area, as well as a master bedroom, two secondary bedrooms, and a bathroom. For reasons which will become apparent during Section 2, key locations within the kitchen environment are labelled in Figure 1, such as the counter-tops, hob/cooker and sink.

A screenshot of a video game

Description automatically generated with medium confidence

Figure 1 - multi-room environmental layout

To model the environment depicted in Figure 1, we idealise the environment as a topological map consisting of a finite number of nodes, , arranged into a graph structure (Figure 2). Each node represents a unique location identified within the environment, with connections occurring between neighbouring nodes, known as edges. Each edge has a unique identifier which describes the identity of the connection, as well as the linear distance of the connecting edge and a risk metric associated with traversing the edge, .

A screenshot of a computer

Description automatically generated with low confidence

Figure 2 - environment node topological map

The distance metric, , represents the Euclidean distance of the edge connecting the two nodes, whilst the the risk metric, , can be considered as an arbitrary value indicating the risk associated with attempting to traverse along this edge. Although the risk metric contains a single numerical value describing the probability of successfully traversing the edge at any instance in time, the framework that initialises the environment computes an additional set of probabilistic parameters, describing the probability of the robot returning to the same location when attempting to move along an edge, or the probability of a complete failure occurring . The values of and for each edge during initialisation of the environment is computed randomly from a normal distribution, such that:

(1)

The risk factor is a measurement of the risk associated with traversing that edge. Although risk is applied in arbitrarily, factors which impact the level of risk include the severity of clutter within the vicinity of the edge, whether the edge intersects one of more other edges, and whether the edge is likely to be in an area which is predicted to be “busy”.

Shape, polygon

Description automatically generated

Figure 3 - variable risk edge connections for environment depicted in Figure 1

Risk is applied to the environmental model based on three primary categories: low, medium, and high. Each category contains an additional sub-category of low, medium, and high, to better refine the environment. For example, in the topological graph environment depicted in Figure 1, node 1 is located at the extremity of the environment and is considered to have a low-busy factor, whilst also having little-to-no intersections with other nodes. This gives transitions occurring from node 1 a “low-medium” risk factor for edges between nodes 1 to nodes 2 and 8. However, since an edge to node 4 would have an intersection, this elevates the level of risk to “medium-low”. On the other hand, node 10 is located centrally in the open-plan living/dining area and has many neighbouring nodes, increasing the busy-factor and elevating the level of risk to high.

# Environment Framework

The framework for initialising and creating a map of the environment has been created programmatically in python and uses object-orientated programming. The class *Graph* takes an input of number of nodes as well as the number of probabilities that form the probabilities of the risk network in Equation 1. The reasoning for this is the same class structure is used to create environmental representations for both the agent and human, but with different risk network structures. The class initialises an object which contains an transition and probability matrix, which is later populated with the environments edge structure.

*agent = Graph(n\_nodes, n\_probabilities)*

The connections are between each of the nodes is dependent on the environment and can be instigated as either a nested list (list of lists) or in json file format. The connections require information relating the start node and the end node for the connection, as well as the linear distance and risk factor. For example:

*connections = [*

*[1, 2, 0.70, ‘low’],*

*[1, 4, 1.20, ‘medium-low’ ],*

*…*

*[27, 28, 0.30, ‘medium-low’],*

*[29, 30, 0.40, ‘medium-low’],*

*]*

Connections are created within the agent object by using the method Create\_Connections() with the connection data structure applied as an input. This will populate the transition and probability matrices, with the method Create\_Map() finalising the environment.

agent.Create\_Connections(connections)

agent.Create\_Map()

# Mission and Task Modelling

A mission is comprised of number of tasks, forming the mission set. Each task consists of a number of sub-tasks, where each sub-task does not necessarily have to be conducted in sequential order, these sub-tasks are denoted as being “unordered.”



*Figure 1 - This is the logo of the ALMI Project*

|  |  |  |  |
| --- | --- | --- | --- |
| Table |  |  |  |
|  |  |  |  |

*Table 1 - This is just a table*