**6.3 Space Discretization - Avoidance Grid**

**Operation Space:** The *Operation Space* is a space where UAS can effectively surveillance its surroundings, and it has the capability to act.

**The motivation for Discretization:** The UAS surroundings needs to be represented in an *avoidance-friendly manner*, following principles matters:

1. *Discrete representation* - the space around UAS should be segmented into finite and exclusive portions which are considered as one point of the grid. This enables fast situation assessment.
2. *Threat proximity* - a threat in any form is getting more important with decreasing distance to UAS.
3. *LiDAR swipe density* - one LiDAR swipe scans many points; the grid needs to be customized to swipe characteristics.

The *Main Sensor* is *LiDAR* (problems **??**.-**??**). The *effective occupancy computation* needs to be done for all problems; the inspiration is taken from [1]. The *effective occupancy computation* is done in *LiDAR* scan portioned into *polar coordinates grid*. The *operation space* is abstracted as a grid where *space portions* are representing the points in the grid.

*Note.* Each member of the grid is a cell, represented as a point with shared properties, like threat level, visibility.

The *Discrete Situation Evaluation* is executed for a *UAS* local coordinate frame in fixed *time*. The goal is to enable *fast discrete situation assessment*.

**LiDAR Swipe:** The *point* scanned by *LiDAR*, where the *UAS position* is center of the *local coordinate frame,* and *UAS heading is defining the main axes* is given as:

*point* = [*distance,horizontal*◦*,vertical*◦]*.* (6.1)

*Note.* For polar/Euclidean transformations and local/global coordinate frames refer to background theory (app. **??**).

The *right side* of UAS *horizontal*◦ ∈ ] − *π,*0[, the *left side* of UAS *horizontal*◦ ∈ [0*,π*], the *downside* of UAS *vertical*◦ ∈ ] − *π,*0[, the *top side* of UAS *vertical*◦ ∈ [0*,π*]

**LiDAR Swipe Portioning:** The *polar coordinate space* can be portioned into distinctive cells, which contains the portion space. This cell then represents one point in the grid.

2

The *reason* for this swipe portioning is *LiDAR* scanning density1, which is extremely dense. The *threat* state in the cell can be assessed with linear complexity.

The *polar* → *euclidean* coordinate frame transformation is not amenable for LiDAR swipe. The *threat* assessment based on *LiDAR swipe* in *planar space portions* has minimal complexity, and it is cost effective. [2].

**Cell:** To discretize operational space into a grid of points there is a need to define cell space, which bounds the portion of the *local planar coordinate frame*. The point (eq. 6.1) is defined by distance, horizontal◦ offset angle, and vertical◦ offset angle. The cell is a closed compact set of such points. The boundary can be defined like follow:

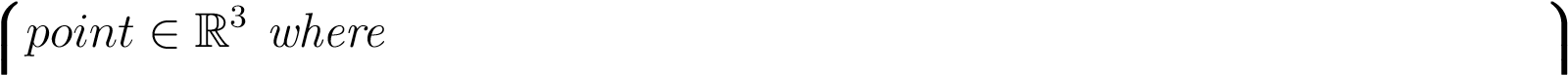
**Definition 1.** *Cell*

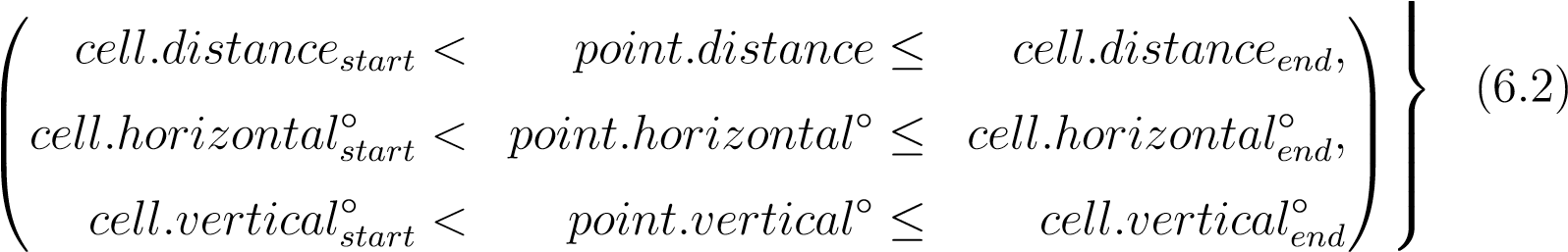
*The* cell *bounds a portion of space in UAS local polar coordinate frame, defined by boundary ranges:*

1. Distance Range *- starts and ends distancestart < distanceend in* R+*.*
2. Horizontal Range *- starts and ends:* *.*
3. Vertical Range *- starts and ends: by* *.*

*The* space portion *belonging to the* cell *is given by function as:*

*cell.spacePortion...*

:



# 

*To evaluate a* static obstacle threat*, it is necessary to know how many LiDAR hits landed in the cell space portion. For one* LiDAR Scan *the* hits set *is given a* set *of* all points *which lands into cell space portion:*

*cell.LiDARHits* = {*point* ∈ *LidarScan* : *point* ∈ *cell.spacePortion*} (6.3)

*Note.* The *cell* space portion volume is increasing with the distance. This satisfies the requirement for threat-distance importance. The cell is considered as a point of the grid with common properties abstraction valid for all cell space portion.

1Example rotary LiDAR Velodyne VL-16 specs: [https://www.cadden.fr/wp-content/uploads/](https://www.cadden.fr/wp-content/uploads/2017/02/Velodyne_VLP-16-Puck.pdf)

[2017/02/Velodyne\_VLP-16-Puck.pdf](https://www.cadden.fr/wp-content/uploads/2017/02/Velodyne_VLP-16-Puck.pdf)

**Effective Operation Space:** The goal is to determine which of the operation space is going to be considered in our avoidance grid. The effective operation space determination according to [3] is influenced by the following factors:

1. *Sensors ranges* - there is no reason to assess the situation over effective *sensor range*.
2. *Information sources* impact - there is no real impact on *effective space boundary*, the information search and intersection algorithms are only of the importance.
3. *UAS maneuverability* - the space where UAS can maneuver, bounded by space-time (reach set boundary).
4. *Computation power* - the situation evaluation and threat assessment capabilities of the onboard computer.
5. *Airworthiness requirements* - the *regulations* can impose some minimal requirements on *effective operation space boundary*.

Let show an example of an *effective operation space* for the UAS (fig. 6.1). The *full LiDAR Swipe* (cyan and red lines) of *UAS* (blue plane) has a *shape* of the conical cylinder.

*Note.* Under *ideal circumstances,* the *LiDAR swipe* would have a *ball shape*, but in real cases the *UAS body portion* where *LiDAR* is mounted is unused.

The *frontal portion* (red line) is a set of cells where *UAS* can make maneuvers. According to the *previous conditions*, there is no reason to consider a space portion out of the maneuverable area.

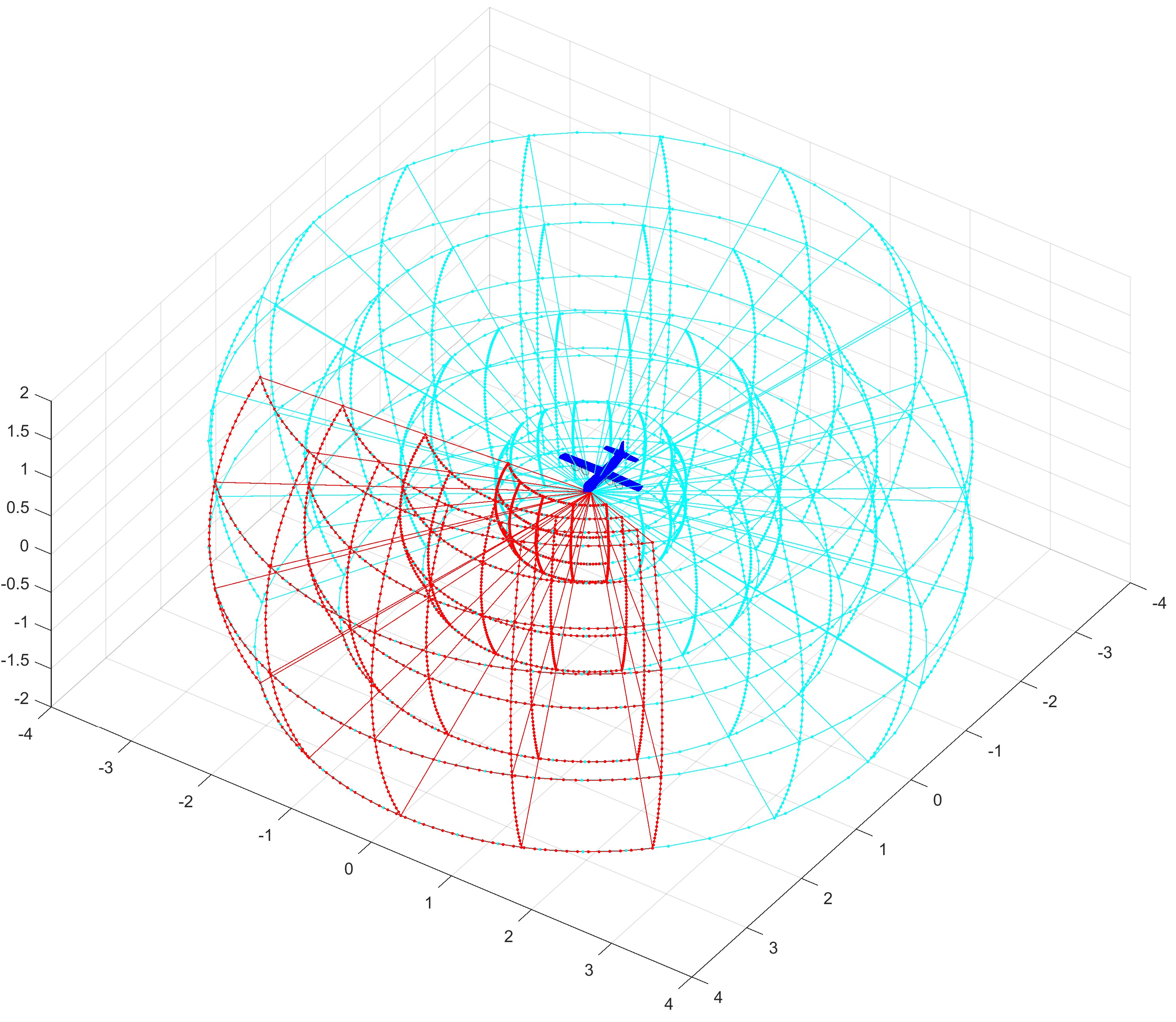


Figure 6.1: Example: The *LiDAR* reading portioning - cells.

4

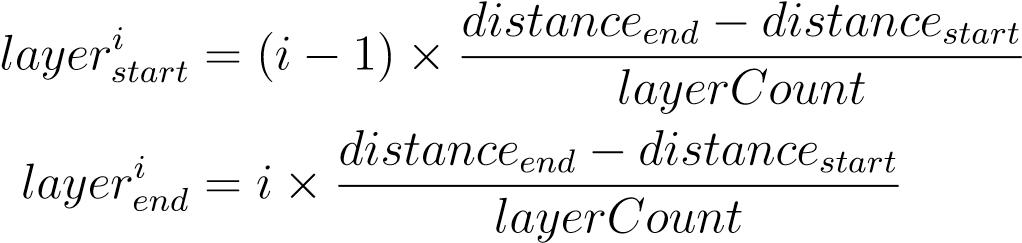
**Avoidance Grid Definition:** The *effective operation space* is going to be portioned into cells. The set of these cells is going to be called *Avoidance Grid*. The idea is to split operational space into cells with even distance, horizontal angle, and vertical angle ranges.

**Definition 2.** *Avoidance Grid*

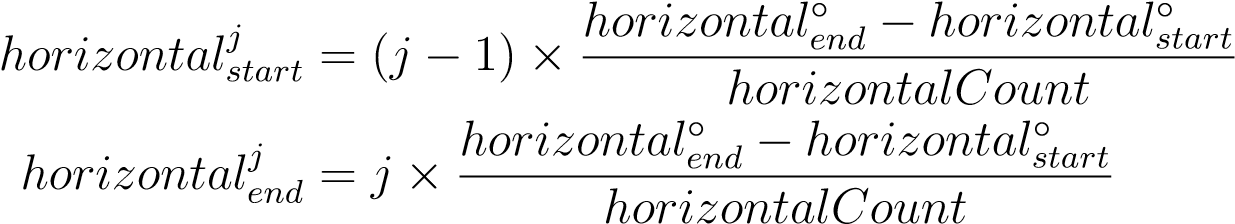
*The* effective space portion *(fig. 6.1 red lines) given by a portion of space in UAS local polar coordinate frame, bounded by:*

1. Distance Range *- in range distancestart < distanceend in* R+*.*
2. Horizontal Range *- in range by**.*
3. Vertical Range *-in range* *.*

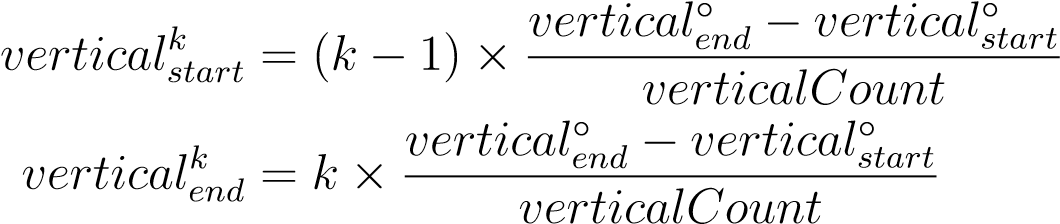
*The goal is to separate the* effective operation space *into cells (def. 1). The idea is to split distance range into multiple distinctive distance ranges with count layerCount* ∈N+*. The ranges for* distance layers *are given as follow:*

; *i* ∈ 1*...layerCount* (6.4)

*The same separation Layer horizontal/vertical separations defined by horizontalCount* ∈N+*/verticalCount* ∈N+*:*

; *j* ∈ 1*...horizontalCount*

(6.5)

; *k* ∈ 1*...verticalCount* (6.6)

*Then celli,j,k space portion by (def. 1) has the following ranges:*

1. Cell Distance Range *(eq. 6.4) depending on layer index i.*
2. Cell Horizontal Angle Range *(eq. 6.5) depending on horizontal angle index j.*
3. Cell Vertical Angle Range *(eq. 6.6) depending on vertical index k.*

*Note.* The example of *Avoidance Grid Cells* is given in (fig. 6.1 red boundary).

*The* Avoidance Grid *is the set of cells:*

  *i* ∈ 1*...layerCount*

#  

*AvoidanceGrid* = *celli,j,k* : *j* ∈ 1*...horizontalCount* (6.7)  *k* ∈ 1*...verticalCount* 

*Note.* For any distinctive cells *celli,j,k*, *cellm,n,o* their *space portion intersection* is empty set:

∀*celli,j,k,cellm,n,o* : *celli,j,k* ∩ *cellm,n,o* = ∅*,i* 6= *o* ∨ *j* 6= *n* ∨ *k* 6= *o* (6.8)

**Grid Sizing Approach:** The sizing approach used in this work is outlined in (app.

**??**).

**Cell in Avoidance Grid Properties:** For each cell *p~* ∈R3 in the there are properties to be checked:

1. *Is there visibility to the cell?* - how good is an observation of the cell by Sensor

Field.

1. *Is there threat present ?* - how sure the data fusion is that there is eminent threat in the cell.
2. *Is the cell reachable?* - if there is any trajectory which can get UAS to that cell without too much threat along the way.

The answers to these questions are given later in *data fusion procedure* outline (tab. **??**). 6

# **Bibliography**

1. Florian Homm, Nico Kaempchen, Jeff Ota, and Darius Burschka. Efficient occupancy grid computation on the gpu with lidar and radar for road boundary detection. In *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pages 1006–1013. IEEE, 2010.
2. Sandeep Gupta, Holger Weinacker, and Barbara Koch. Comparative analysis of clustering-based approaches for 3-d single tree detection using airborne fullwave lidar data. *Remote Sensing*, 2(4):968–989, 2010.
3. Osmar R Za¨ıane and Chi-Hoon Lee. Clustering spatial data when facing physical constraints. In *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*, pages 737–740. IEEE, 2002.

7