Problem setting and definitions

July 10, 2018

The problem we address deals with a team of robots exploring an environment initially unknown. The aim of this work is to find out if some features of the environment can be exploited in order to improve the exploration performances. This exploration problem can be formally defined through a 4-ple $\langle A, E, P_0, T \rangle$ where:

- A is the set of robots used for the exploration;
- E is the environment to explore;
- $P_0$is a vector of the initial poses of the team;
- T is the termination criteria.

This four elements characterize the instance of exploration problem we are addressing, while the solution of the problem consists in producing a map M of the environment E. The map M consists in a 2D occupancy grid representing the areas of E. The center of each cell c is characterized by its coordinates in the environment, provided as a vector (x_c, y_c). Each cell contains a variable, whose value tells if that cell is clear, occupied by an obstacle or still unexplored. Clearly, the first two values refer only to already explored cells, while the third refers to those cells which are still not mapped by any robot of the team. In particular, a cell is clear if there are no obstacles in it, otherwise, it is occupied. As stated above, the aim of this work is to find out if it exists a correlation between a set of features of the environment and a possible improvement in the exploration performance. The features taken into account are:

1. Dimension: small environments will take less to be explored with respect to large ones
2. Openness: the property of an environment to be composed of large spaces rather than cluttered ones
3. Parallelizability: a measure of how much the robots are enforced to spread during the exploration or if they will stick together

Being the environment unknown, we cannot use it directly to extract these features. Thus, we need to use the map produced by the robots for teh computation and this feature extraction step should be done as the exploration goes on; in this way, we can update the coordination strategy online. For this reason, we define as M_t the map known at time $t \in [t_0, t_f]$ where t_0 is the time at which the exploration starts and t_f is the time at which the exploration ends, i.e. the time at which the termination criteria T is verified. We also define the auxiliary function $S : M \to \mathbb{R}$ which takes as input a cell of the map and provides its area.

Dimension and openness computation can be easily defined being rather intuitive features of an environment. On the other hand, parallelizability needs further considerations.

Dimension can be computed as the amount of free area in the map, thus it is the sum of the area of all the clear cells: $D(M) = \sum_i S(f_i)$ with f_i being a clear cell.

Openness is related to the obstacle density, so to the probability of a randomly picked cell to be occupied by an obstacle: $O(M) = \frac{\Sigma_i S(o_i)}{\Sigma_j S(c_j)}$ where o_i is an occupied cell and c_j is a generic cell.

Before moving to parallelizability, we need to introduce the notion of embedded graph. A graph G embedded to a surface Σ is a representation of G on Σ such that points of $\Sigma$and simple arcs in it are associated with vertices and edges of G respectively. In the case we are analyzing, this concept can be exploited by creating a graph Υ embedded on the map M and then through the concept of centrality of a node, we can find out what are the most connected nodes of Υ. If a node has a lot of connections, it means that it is a splitting point for the team and this information about the node is very important for the coordination management. On this setting, we can define the parallelizability according to the number of central node in the graph and on the average number of connections of each node.