Robot’s Localization using a Map of landmarks

Localization

The term localization means the estimation of the position and orientation (aka “pose”) of the robot. There are diverse approaches and sensing technologies which can contribute to achieve the localization of a robot.

Localization, why? Answer: The robot needs to know its position and orientation, in order to interact with the environment. In almost all real applications the robot is not allowed to be “lost” (i.e. a situation in which the robot does not know its position.)

Ideal Localization

It would be an ideal situation if the robot were able to know its pose, at any time and place, by just reading that information from a sensor. Such a fully reliable sensor does not exist in the reality (at least at present). In order to guarantee the estimation of the pose, at any time, the robot should perform fusion of information from different sources. Usually, there is a process which performs the localization task, and offers the results to other processes which run in the robot (such a process behaves as a virtual sensor, for the rest of client processes).

There are diverse approaches for obtaining estimates of the position. The GPS (Global Positioning System) is a well known one (e.g. GPS and GLONASS systems). However, its estimates are not perfect, and it is not available at certain places (due to occlusion, multipath problems or in contexts such as indoor, underwater and underground environments, etc.).

GPS can be improved by combining its measurements with other sources of information, such as Inertial Systems, speed meters, etc. The result of that fusion process provides more accurate and reliable estimates about the position.
Inertial Systems by themselves are not designed to operate in a long-term fashion. They measure and integrate accelerations and angular velocities. Minor errors due to bias and other uncertainties could have remarkable effect, when they are integrated for obtaining the position estimates. Except the very expensive inertial systems used in expensive equipment (submarines, etc.), they are not intended to provide the solution for the long-term localization problem.
Compasses can provide measurement of the orientation of the machine. However, these measurements are affected by the distortion of the Earth’s magnetic field, due to the presence of infrastructure or by electric equipment (motors, generators, transformers, power lines, etc.).
One interesting option for the robot’s localization is the utilization of a “localization map”, jointly with on-board sensors, to estimate the position of the robot in that map by matching certain map features with features observed by the robot’s perception. A navigation map is a map that describes a set of known landmarks. Landmarks are objects which are clearly detected and identified by the robot. If the robot can detect a landmark and can measure the relative position of that landmark (in respect to the robot coordinate frame), and, if it also knows the absolute position of that landmark (provided by the

navigation map), then the robot may use that information to improve its knowledge about its position. If a sufficient number of landmarks are detected in that way, then the robot can perform a *triangulation* process to estimate its position.

Map Based Localization

Approaches that use a map of landmarks and on-board sensing capabilities of the platform (for detecting those landmarks) are widely used in Field Robotics. Different sensors produce different type of information. Some sensors detect the presence of objects and measure the range, the bearing or both. In addition, other properties can also be measured (color, shape, texture, etc). Those additional properties are useful for inferring the identity of the landmarks (a problem called "Data-Association".)

Range Only

In this case, the measurements provide the distance between a point in the vehicle body (usually the position of a scanning sensor) and some known landmark. Sonar measurements are a good example of a range-only sensor. Another example of a process performing Range-Only triangulation is a GPS receiver. The GPS estimation process is implemented (by the GPS receiver's software) based on range-only measurements (each range is based on the time of flight of a RF signal between a satellite and the receiver's antenna); in this case the satellites play the role of the landmarks. By combining the range measurements (each of them related to an individual satellite), the GPS receiver can estimate the position of its antenna, in a global coordinate frame. If enough range measurements are available (i.e. enough number of visible satellites) the GPS receiver can also estimate its clock's errors (GPS receivers' clocks are low cost clocks, not very accurate, and time measurement is crucial for converting Time of Flight measurements (ToF) to range).

A range measurement implies a constraint in our belief about the robot's pose coordinates. It means that the robot must be at a certain distance from the landmark, i.e. in a 2D environment the constraint defines a 1D manifold, which is a circumference ➔ the position of the robot is one of the infinite points on a circumference of radius=Range (measured range) and center at the landmark position.
(See whiteboard during lecture!)

Equations:
Each measurement introduces an equation (constraint) of the form:

$$r_i = \sqrt{(x_i - x)^2 + (y_i - y)^2} \ ,$$

in which the unknown variables, (x, y), are the coordinates of the position of the robot; (x_i, y_i) is the known (and usually constant) position of the landmark number i , and r_i the range measured, i.e. the measured distance between the current robot's position and that particular landmark.
Just one constraint like this one is not sufficient for fixing the position estimate, i.e. there are infinite possible solutions (points) that satisfy the constraint (so we are still uncertain about the robot's position).

When the platform can simultaneously detect multiple landmarks, the associated measurements introduce a set of equations which allows the estimation of the position of the platform. Now the solution must satisfy many constraints, simultaneously; i.e. mathematically expressed: the unknowns (x, y) must satisfy a set of equations.

$$\left\{ \begin{array}{c} r_1 = \sqrt{(x_1 - x)^2 + (y_1 - y)^2} \\ .. \\ r_n = \sqrt{(x_n - x)^2 + (y_n - y)^2} \end{array} \right\}$$

Where the items $x_1, y_1, x_2, y_2, .., x_n, y_n$ are constants known in advance (provided by a navigation map) and the items $r_1, r_2, .., r_n$ are the range measurements, provided by the platform's on board sensing resources. The variables (x, y) are initially unknown; however, they can be estimated by solving the set of equations.

More than three equations may seem to be redundant as the three circumferences must intersect at the solution point. However, in real cases, the presence of uncertainty in the measurements and the presence of certain uncertainty in the knowledge of the landmarks' positions usually make the "redundancy" a necessary situation in order to reduce the uncertainty in the estimated position of the robot.
In many cases just two observations are enough if the robot has some a-priori estimation of its position in order to solve the ambiguity in the solution of the equations (the circumferences do intersect at two points ➔ there are feasible two solutions, but we choose the one that is the nearest to our prediction.)
There are pathological cases in which both curves do intersect at just one point (unusual case), and even cases in which no intersection does happen at all, we discuss about it, in class.

(Space for figure: See whiteboard during lecture)

Based on this naive approach, range-Only measurements cannot provide estimates of the heading of the platform (i.e. the distance between the platform and a landmark is independent of the heading of the platform). This means that heading is not *observable* by just doing triangulation (However, by repeating this process at different points and by combining the observations with the kinematic model of the robot, it is possible to estimate the heading. This concept is not studied in MTRN4110).

Bearing only Observations

This type of measurements provides the angle of the vector defined by the machine's position and the landmark's position. Typical sensors for this type of measurement are cameras. Each pixel in the image is related to a certain azimuth and elevation angle. In a 2-dimensional case, the angle considered is the azimuth. Each measurement of a landmark introduces an equation which involves the 3 robot's pose variables (x, y and heading),

$$\theta_i = \arg\left(\left(y_i - y\right), \left(x_i - x\right)\right) - \varphi + \frac{\pi}{2}$$

in which θ_i is the measured angle (azimuth) and φ is the heading of the machine. The function $\arg\left(y, x\right)$ is the argument of the vector $\left(\mathrm{x}, y\right)$. This expression can present some modifications, depending on the convention used for expressing the heading. See the convention we use in the lecture (you may use a different axis and angle convention, so you would need to properly adapt the equation).

(Space for figure: See whiteboard during lecture)

Note that just one observation (i.e. one constraint) tells us that the possible solutions are in a 2D surface in the 3D domain (x, y, heading). Any of the infinite possible solutions on that surface is possible.

If multiple landmarks are observed simultaneously, then a system of equations (constraints) is defined, as the following one,

$$\begin{Bmatrix} \theta_1 = \arg\left(\left(y_1 - y\right), \left(x_1 - x\right)\right) - \varphi + \dfrac{\pi}{2} \\ .. \\ \theta_n = \arg\left(\left(y_n - y\right), \left(x_n - x\right)\right) - \varphi + \dfrac{\pi}{2} \end{Bmatrix}$$

Just three equations (assuming they are independent (*)) are enough for solving the system, to obtain the full pose of the machine (i.e. $\left(x, y, \varphi\right)$).

Usually, there is uncertainty polluting the measurements of the angles and in the knowledge of the landmarks' position; consequently, more constraints are usually a good way to improve the estimation of the unknowns $\left(x, y, \varphi\right)$.

(*) if we repeat the same equation or we observe landmarks that are very close between each other, then they would not provide enough information for implementing a reliable triangulation process (a "ill conditioned problem").

Range and Bearing Measurements

Some sensors can provide range and angle measurements of the objects. Examples of these sensors are radar, laser scanners, stereo vision, 3D cameras, etc. Each of those sensing technologies presents different performance in terms of quality (uncertainty in range or angle measurements), sample rate, range and cost.
In 2D localization, a measurement that includes range and bearing, introduces two constraints (equations) for each observed landmark. The related constraints can be expressed in polar representation, i.e. range and bearing, as shown in the following equations:

$$\begin{cases} r_i = \sqrt{\left(x_i - x\right)^2 + \left(y_i - y\right)^2} \\ \theta_i = \arg\left(\left(y_i - y\right), \left(x_i - x\right)\right) - \varphi \end{cases}$$

At least two observations of the type range-bearing are needed to obtain the pose of the machine (i.e. these observations would introduce four equations, which would be sufficient for obtaining the three unknowns).
In 3D, range and bearing means the range and two angles (elevation and azimuth). In 3D localization the pose estimation involves six unknown variables (x, y, z and three attitude Euler angles, e.g. roll, pitch and yaw.)

Data Association

In addition to the measurement and the a-priori knowledge about the map (i.e. the positions of the landmarks) the machine needs to identify the landmarks. For example: "*I am observing a pole, however there are 20 surveyed poles in this area of operation; which one is this?"*. If I knew which pole is the pole that I am observing, then I would be able to generate the equation (or equations) associated with the observation. Doing the wrong data association (i.e. assuming the wrong pole ID) ➔ will produce inconsistent results (I will estimate a wrong position!)
There are methods (e.g. in stochastic data fusion) which can deal with multi-hypotheses (Situations such as "this pole may be pole 5 or pole 17"). Even under those ambiguous situations, those methods are able to extract information, for finally solving the ambiguities. Those methods are not discussed in MTRN4110.

How to perform Data-Association if all the landmarks look the same to me?

The process of localization is not performed in just one step. As the robot moves it keeps continuously receiving information from observations of landmarks and from its process model (kinematic model). If at time t_1 the robot has an estimate about its pose, then for some time later, $t_2 = t_1 + dt$, it is possible to predict the position at that time. The quality of the predicted new pose would not be good for localization purposes; however, it would usually be accurate enough for data association purposes.
In our projects, we implement the triangulation process based on range and bearing measurements. Consequently, for solving the Data Association we can apply the following procedure:
We have the last estimated robot's pose, then, by applying the kinematic model of the robot, we predict the current pose. Based on the predicted current pose and on the range and bearing measurements, we predict the global position of the locally detected landmarks. i.e. for each detected object we use its measured local position (expressed in the robot's coordinate frame) and express it globally (by properly rotating and translating its coordinates). This is possible because we approximately know the current robot's pose. Once we have the predicted global position of the detected landmark, we compare it against the list of landmarks of the navigation map; we are then able to infer the identity of the detected landmark. Those detected landmarks that we successfully identify can be used in our triangulation steps; which improves our estimates about the robot's pose.
This process keeps repeating periodically, performing the three steps: prediction, data association and updates.

Next document to read: Applying EKF for solving the map-based localization.