# Executive Summary

In this project, we have focused on robust estimation algorithms that can be used for underwater position estimation (localization) from range data with ambiguity, such as GPS or sonar signals. We were able to formulate this problem as graph-SLAM and solve it using the associated probabilistic framework.

We have implemented several variants of range-based localization for different modalities of the problem. We have implemented motion models to be used to help reducing the ambiguity in the localization problem.

Several experiments were performed and we also recorded a corpus of differential GPS data for localization testing, at LKKOTV Kotvrdovice airport.

In the project, there was researcher mobility, both Jean-Marie Codol visiting BUT and Viorela Ila visiting BOT in France for integration of the developed software.

## Project progress

Objective name: Development of graph representation of range-based localization problems

Objective type: Research

Objective description: Define edge and vertex types for graph-SLAM-like optimization of RO localization.

Impact: Being able to localize using MLE class of algorithms more precisely and with less computation than if using Kalman filtering approach.

Status: Finished, developed formulation and experimentally validated in Matlab. Some synthetic benchmarks were also generated, for the initial algorithm testing.

Objective name: Collect experimental GPS data

Objective type: Data collection, implementation

Objective description: Implementation of GPS loggers in hardware and software, in order to be able to record GPS data, which in itself Is related to range-only localization, and has reasonable ground truth available from the conventional receivers.

Impact: Data collection for further testing and benchmarking.

Status: Built a pair of SIRF-3 hardware dongles using SIRF-3 modules and FTDI-232 chip, and implemented data logging software for Linux and Windows. Flew and recorded several hours of data in a personal sports airplane at LKKOTV Kotvrdovice airport.

Objective name: Development & testing of fast localization library

Objective type: Implementation

Objective description: Implementation of the algorithms developed in Matlab, in C++ and subsequent optimization for vector processors using SSE and also NEON instruction sets for embedded platforms.

Impact: Being able to use the algorithms developed in a practical implementation, and in the hardware platform developed by BOT in France.

Status: The algorithms were implemented as a plugin for the SLAM++ fast optimization library, and integrated with BOT.

Objective name: Implementation of a visual demonstrator

Objective type: Implementation

Objective description: Implementation of a simple visual demonstrator that could be used for PR purposes by BOT.

Impact: Being able to showcase the technologies developed, in easily understandable manner.

Status: Implemented a simple OpenGL ES visualization of the localization algorithms, which can be used on a PC or on tablets.

## Deliverables

The Faculty of Information Technology from Brno University of Technology (FIT BUT) provided a study of feasibility and efficient solutions to several problems encountered in an underwater positioning system.

* *BOT* GPS Intelligent Buoys-based (GIB) underwater positioning system was formulated as a graph optimization problem.
* *SLAM++*, the nononlinear optimization software developed at BUT was extended to work with the new types of variables (robot’s trajectory and position of the buoys) and constraints (measured signal from the buoys). SLAM++ is an efficient implementation of incremental nonlinear least squares solvers. It is written in C++ and it is very fast due to the fact that it exploits the block structure the problems and offers very fast solutions to manipulate block matrices within iterative nonlinear solvers.
* Several data acquisition was performed and used to test the implementation.
* Several hardware platforms considered in the tests:
1. SLAM++ was tested on Raspberry Pi, Model B (the one with 512 MB of RAM), this being the platform chosen for the implementation of the GIB system. A tutorial about how to install, compile and run SLAM++ on Raspberry Pi architecture was provided.
2. SLAM++ was extensively tested on CPU platforms and the results were reported in papers published in top international conferences. Real-time performance was obtained for robotic localization systems.
3. A GPU implementation of the SLAM++ is currently under development.
* An extensive SLAM++ documentation is provided. The documentations contains:
1. Instructions on how to compile and run the code on different platforms and operating systems
2. The description of the functionality of the code
3. Examples of applying the SLAM++ for solving 2D and 3D robotic localization problems
4. Instructions on how to run your own application using SLAM++