

Local map update for large scale SLAM

J. Aulinas, J. Salvi, X. Lladó and Y. Petillot

A technique for simultaneous localisation and mapping (SLAM) for large scale scenarios is presented. This solution is based on the use of independent submaps of a limited size to map large areas. In addition, a global stochastic map, containing the links between adjacent submaps, is built. The information in both levels is corrected every time a loop is closed: local maps are updated with the information from overlapping maps, and the global stochastic map is optimised by means of constrained minimisation.

Introduction: Simultaneous localisation and mapping (SLAM), also known as concurrent mapping and localisation (CML), is one of the fundamental challenges of robotics. The well known strategy of using the extended Kalman filter (EKF) suffers from computational and consistency problems mapping large environments. Many solutions have been proposed to reduce EKF cost. Among them, the most promising ones are those that split the whole area into small submaps. For instance: the Map Joining algorithm (MJ) [1], which builds submaps of a limited size to join them in a global map; the Divide and Conquer algorithm (D&C) [2], which joins local maps following a binary tree hierarchy; the Atlas approach [3], which builds a graph with the transformations between adjacent local maps. This graph allows us to relocate each local map with respect to any other local map. Using a map matching algorithm, cycles in the graph are verified and optimised; the Hierarchical SLAM algorithm (HS) [4], which not only optimises the global level by means of loop closing constrained optimisation, but also updates local map information by joining those local maps that are closing the loop. All these algorithms use EKF based SLAM to build submaps, but they all differ in the way they manage these local maps at a global level. At this global level, the use of graph theory tools to decide relevant links to closed loops is a very interesting idea. In this Letter we present a new SLAM strategy that benefits from these graph theory tools at the global level, while locally the use of common feature estimates from overlapping maps are used as observations to revise the local map information. These corrected submaps are not joined to any other submap, instead they are maintained separately, keeping its original size, and benefiting from the limited size of the matrices involved during global and local update processes, reducing the computational demand without paying any consistency decrease.

Submaps and stochastic map building: Maps of limited size are built by means of a standard EKF based SLAM algorithm. This is possible under the assumption that an EKF algorithm is able to perform consistently when dealing with small maps. The size of a map is limited by different parameters that need to be tuned, such as the total number of features within the local map, the relative displacement of the vehicle from the very beginning of the local map to its end, or boundaries for acceptable uncertainty levels. Every submap is initialised with the vehicle position at the coordinates origin as the base reference of the map, with no uncertainty. The vehicle moves through the scene and thanks to its onboard sensors, odometry measurements are obtained as well as the position of environment features relative to the vehicle current location. This information is first evaluated on a data association algorithm and then used in the filter to update both the map and the localisation of the vehicle within it. During the data association process outliers are rejected, for instance, moving objects or false detected features. This submap building process ends when the submap reaches the limitations, resulting in a map M_i with its state vector x_i and its uncertainty P_i . Afterwards a new submap is started. Compared to other submap techniques, our approach does not share any information between consecutive maps, instead they are kept completely independent. Meanwhile, a global stochastic map is built as in [3, 4]. This global level is a graph that has all the submaps built so far as nodes, and its arcs represent the topological link between adjacent maps. These links are the relative transformation T between submaps' bases $x_u = (T_2^1, T_3^2, \dots, T_n^{n-1})^t$ and its associated uncertainty P_u .

Loop event, global and local update: Every time a local map is completed, its base reference is referred to the global frame, and its distance to other maps' bases is computed. Those maps that lie near the last submap are identified as possible loop closure candidates. Taking

advantage of the links provided by the global level graph, the last submap can be referred with respect to any of the listed maps' base reference. Therefore, its overlapping can be computed by running a data association routine. In front of a loop closing event, some features from these maps will be associated, and the loop hypothesis will be accepted. In addition, the overlapping between the last map and the previous one is checked in order to avoid closing loops between successive maps. Besides, if both maps have a high degree of common information, both are corrected using the local map update procedure. Every time a loop event is detected, the global stochastic map is recomputed by means of constrained minimisation [4]:

$$\min f(x) = \min \frac{1}{2} (x - x_u)^t P_u^{-1} (x - x_u)$$

$$h(x) = T_2^1 \oplus T_3^2 \oplus \dots \oplus T_n^{n-1} = 0$$

where x is the stochastic map after being constrained. This constraint forces the composition of all the transformations between maps' bases involved in the loop to be zero. Once the links between submaps have been recomputed, a local map update procedure is performed. The conceptual idea is to use the information from overlapping maps, found to be closing a loop, to correct themselves. The basis of this procedure is to consider those common landmarks from one map as the predictions of a standard EKF prediction stage, while the common features from the rest of the overlapping maps are understood as new measurements. From them the innovation vector and the EKF gain are calculated and an EKF update is performed. This routine is executed for each overlapping local map. This double correction may lead to overconfidence, which means that the vehicle could end up completely lost. In order to solve this issue, process noise is added to this procedure.

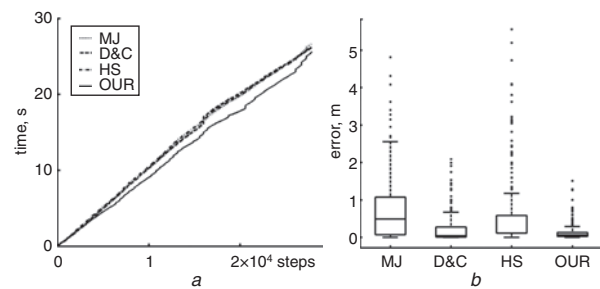


Fig. 1 Result comparison

a Computation time against filter step (step = 25 ms)
b Consistency boxplots

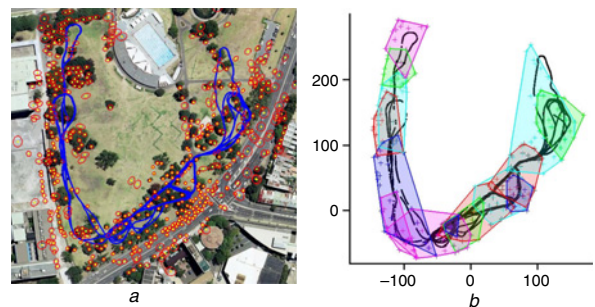


Fig. 2 Final map

a Satellite image of Victoria Park
b Submaps build during process

Results: In order to evaluate our approach we use the Victoria Park data set recorded by Eduardo Nebot (<http://www-personal.acfr.usyd.edu.au/nebot/dataset.htm>) at the Australian Centre for Field Robotics. This data set describes a path through an area around 197 × 93 m. It consists of 7247 frames along a trajectory of about 4 kilometres length, recorded over a time frame of 26 minutes. The data set contains sensor readings from steering and rear-axis wheel (odometry) and laser range finder (one 360° scan per second) along with the ground truth position data from GPS. For the laser range data a tree detector function is provided together with the dataset. The trees are used as features for the SLAM algorithms. They usually have a large distance to each other and can be uniquely identified with common data association techniques. In

order to provide a quantitative evaluation of the results, we compare our approach with the MJ, D&C and HS approaches. More specifically, we evaluate and compare the time consumption and the map consistency (see Fig. 1). In order to do so, synthetic trajectory and landmarks are created on the Victoria Park dataset, which are then used as a ground truth. Fig. 1a shows the computation time along the whole Victoria Park experiment. Observe that our approach requires less computational time. Fig. 1b summarises the error in boxplots committed at the end of the experiment. Note that our proposal has lower uncertainty values. Moreover, the mean, lower and upper quartile ranges are more compact. Finally, Fig. 2 shows the final map, where our final mean error was 0.30 m.

Conclusions: In this Letter a novel SLAM approach suitable to map large scale scenarios is presented. The two main differences with respect to other methods that use submaps are: 1) Local maps are kept independent during the whole scenario, which is shown to be positive in terms of computational cost. The results demonstrate that the total time required to navigate the whole scenario is shorter compared to other methods. 2) Local maps are updated every time a loop is closed reducing the final uncertainties.

Acknowledgments: This work has been supported by the Spanish Ministry of Science and Innovation (DPI-2007-66796-C03-02). J. Aulinas holds a UdG-2007 grant.

© The Institution of Engineering and Technology 2010
5 August 2009

doi: 10.1049/el.2010.2271

One or more of the Figures in this Letter are available in colour online.

J. Aulinas, J. Salvi and X. Lladó (*Institute of Informatics and Applications, University of Girona, Girona, Spain*)

E-mail: jaulinas@eia.udg.edu

Y. Petillot (*Ocean Systems Lab, Heriot Watt University, Edinburgh EH14 4AS, United Kingdom*)

References

- 1 Castellanos, J.A., Martínez-Cantón, R., Neira, J., and Tardós, J.D.: 'Robo-centric map joining: Improving the consistency of EKF-SLAM', *Robot. Auton. Syst.*, 2007, **55**, (1), pp. 21–29
- 2 Paz, L.M., Tardós, J.D., and Neira, J.: 'Divide and conquer: EKF SLAM in $O(n)$ ', *IEEE Trans. on Robotics*, 2008, **24**, (5), pp. 1107–1120
- 3 Bosse, M.C., Teller, S., Leonard, J.: 'ATLAS: A Framework for Large Scale Automated Mapping and Localization'. Ph.D. Thesis, Massachusetts Institute of Technology, 2004
- 4 Estrada, C., Neira, J., and Tardós, J.D.: 'Hierarchical SLAM: real-time accurate mapping of large environments', *IEEE Trans. Robot. Auton.*, 2005, **21**, (4), pp. 588–596