Efficient Construction of Optimal and Consistent Ladar Maps Using Pose Network Topology and Nonlinear Programming

Alonzo Kelly and Ranjith Unnikrishnan

Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3890,
alonzo@ri.cmu.edu

Abstract. Many forms of mobile robot guidance map can be visualized as a network of nodes (images) connected by edges (poses). We show how explicit reasoning about the loop topology of this pose network can generate a cycle basis for the network in linear time. This basis can then be used to form the independent constraint equations in a natural constrained optimization formulation of the mapping problem. In many practical cases, the pose network has sparse structure, and the associated equations can then be solved in linear time. This approach can be used to construct optimal, consistent maps on very large scales in very limited computation. While the technique is applicable to mapbuilding in general, and even optimization in general, it is illustrated here for batch processing of 2D ladar scans into a mobile robot guidance map.

Introduction

Mapbuilding is an important problem in mobile robotics because triangulation from some form of map is often the only mechanism that enables stable accurate position estimates over either long time periods or long excursions. Regardless of the sensing modalities used, the environmental features used as landmarks, or the dead reckoning systems used to compute position between fixes to the map, the construction of a map which is sufficiently accurate and consistent remains a difficult problem.

This paper addresses the following problem. A sequence of ladar scans has been captured during a mapping pass through the environment. The imagery is tagged with estimated vehicle or sensor positions computed from all available dead reckoning indications, potentially including ladar scan matching (visual odometry). Based on these position estimates and an indication of which nonconsecutively captured scans overlap, a globally consistent map must be constructed.

Prior Work

This work is an adaptation of our earlier work on navigating from globally consistent appearance mosaics [5]. Building a map is closely related to the computer vision problems of registering range scans to produce a geometric model or to registering imagery to produce mosaics.

The processing of large amounts of imagery into consistent representations was a core pursuit of analytic photogrammetry before there was computer vision or even computers [16]. The formulation of mosaicking (then known as block triangulation) as that of minimizing the total map residual using the normal equations was a basic technique. In the contemporary mosaicking literature, [4] is similar to the present work in its use of topological information to guide the optimization process. In the range image processing literature, we cite [8] and [9] for pioneering the iterative closest point algorithm, and [10] for using it to simultaneously minimize the total residual error of a number of overlapping range scans.

The idea of a network of reference frames connected by uncertain relative poses goes back in robotics at least to [12]. Many of the core concepts of consistency and constraint in uncertain geometric networks were introduced to robotics in [17]. These, in-turn, have a partial basis in classical network theory [18].

The fact that tracking covariance leads to quadratic complexity in SLAM has been known since [11]. The fact that correlations must nonetheless be tracked was shown in [14]. These results clarify the value of our efforts to diagonalize covariance here. The fact that relative poses can do this was pointed out in [3].

One of the important aspects of this work is its ability to produce maps on the very large scale. In part, this is enabled by an inherently efficient formulation. Highly successful efforts to do the same for incremental approaches are also ongoing [13] [15]. In contrast to the hierarchical decomposition approach of the latter, our approach allows every image in the map to move in order to seek the constrained optimum but otherwise, the use of paths through the network for formulating constraints on relative poses makes this work very similar.

Within the area of incremental mapping, global consistency of range scans was enforced in [2] by solving an overconstrained system of measurements to compute the required least-squares perturbation to absolute scan poses. This approach is extended in [1] to perform online mapping in cyclic environments. In contrast to these highly relevant works, our approach avoids the cubic complexity of the inversion of fully populated matrices in order to produce a method which is practical on the very large scale.

Constrained Optimization

Constrained Optimization in a Relative Map

Consider a representation of the mapping problem in terms of relative poses. In principle, if there are n images, up to n^2 relative poses can be defined. The relative pose of image i with respect to image j will be denoted by $\underline{\rho}_i^j$. In practice, only those relative poses for which an important overlap exists need to be considered.

Consistency relations in this formulation are hard constraints on the relative poses. In general, using T to denote a homogeneous transform, internal consistency is enforced by loop equations which take the form:

$$T_{i+1}^{i} T_{i+2}^{i+1} \cdots T_{i-1}^{i-2} T_{i}^{i-1} = I \tag{1}$$

Such equations require the total transform around a closed loop to be the identity. Likewise, consistency with external standards may be imposed with path equations of the form:

$$T_{i+1}^{i} T_{i+2}^{i+1} \cdots T_{j-1}^{j-2} T_{j}^{j-1} = T_{j}^{i}\Big|_{\text{external}} \tag{2}$$

A loop is simply a closed path in these expressions and the path is therefore the more general notion. Implementations can treat loops as special paths. Path constraints are the mechanism by which external survey information can be incorporated into the map building process.

Let the state vector $\underline{x}$ denote the vector of all relative poses for which an important overlap exists. The squared total residual $\underline{z}(\underline{x})$ of matching points or features in image pairs serves as a useful measure of performance for optimization purposes. The problem specification is.

$$\begin{aligned} &\text{minimize:} && f(\underline{x}) = \frac{1}{2} z(\underline{x})^T \underline{z}(\underline{x}) \\ &\text{subject to:} && \underline{c}(\underline{x}) = \underline{b} \end{aligned} \tag{3}$$

Such problems are, of course, solved using Lagrange multipliers. We adjoin the constraints to the performance index to form the Lagrangian:

$$l(\underline{x}, \underline{\lambda}) = f(\underline{x}) + \underline{\lambda}^T [\underline{c}(\underline{x}) - \underline{b}] \tag{4}$$

and require of a constrained minimum that:

$$\begin{aligned} \frac{\partial l}{\partial \underline{x}} &= \frac{\partial f}{\partial \underline{x}} + \underline{\lambda}^T \frac{\partial \underline{c}}{\partial \underline{x}} = \underline{0}^T \\ \frac{\partial l}{\partial \underline{\lambda}} &= \underline{c}(\underline{x})^T - \underline{b}^T = \underline{0}^T \end{aligned} \tag{5}$$

To generate the constrained Newton's method, we linearize these equations

about a point where they are not satisfied, transpose where convenient, and require satisfaction upon perturbation. The ultimate result of this manipulation is:

$$\begin{bmatrix} \frac{\partial^2 l}{\partial \underline{x}^2} & \frac{\partial \underline{c}}{\partial \underline{x}}^T \\ \frac{\partial \underline{c}}{\partial \underline{x}} & 0 \end{bmatrix} \begin{bmatrix} \Delta \underline{x} \\ \Delta \underline{\lambda} \end{bmatrix} = - \begin{bmatrix} \frac{\partial l}{\partial \underline{x}}^T \\ \underline{c}(\underline{x}) - \underline{b} \end{bmatrix} \qquad (6)$$

Where

$$\frac{\partial^2 l}{\partial \underline{x}^2} = \frac{\partial^2 f}{\partial \underline{x}^2} + \underline{\lambda}^T \frac{\partial^2 \underline{c}}{\partial \underline{x}^2} \qquad \frac{\partial l}{\partial \underline{x}} = \frac{\partial f}{\partial \underline{x}} + \underline{\lambda}^T \frac{\partial \underline{c}}{\partial \underline{x}}$$

Approximating the Hessian leads to:

$$\begin{bmatrix} H^T H + \mu P_o^{-1} & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} \Delta \underline{x} \\ \Delta \underline{\lambda} \end{bmatrix} = - \begin{bmatrix} H^T \underline{z} + C^T \underline{\lambda} \\ \underline{c}(\underline{x}) - \underline{b} \end{bmatrix} \qquad (7)$$

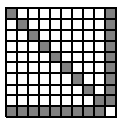
These equations can be iterated until convergence from a initial estimate in order to produce the constrained local minimum. The equations can be solved by recursive partitioning to produce a more efficient form.

$$\begin{aligned} C(H^T H + \mu P_o^{-1})^{-1} C^T \Delta \underline{\lambda} &= (\underline{c}(\underline{x}) - \underline{b}) - C(H^T H + \mu P_o^{-1})^{-1} C^T (H^T \underline{z} + C^T \underline{\lambda}) \\ (H^T H + \mu P_o^{-1}) \Delta \underline{x} &= -(H^T \underline{z} + C^T \underline{\lambda}) \end{aligned} \qquad (8)$$

Complexity of the Optimized Relative Map Formulation

The first equation is solved for the updates to the Lagrange multipliers and the second is then solved for the state perturbation. With careful implementation, the complexity of these computations is $O(m + nl + nl^2 + l^3)$ for the first equation and $O(m + nl)$ for the second. The highest power of n that appears is unity. As shown in figure 1, the matrix to be inverted in this case is block diagonal. The relative poses are decorrelated from each other.

$$\begin{bmatrix} H^T H + \mu P_o^{-1} & C^T \\ C & 0 \end{bmatrix} =$$



shaded cells denote nonzero elements

Figure 1: Closing a Single Loop With Nonlinear Programming. The diagonal part of the matrix is all that needs to be inverted.

It is the decision to use a representation of potentially inconsistent relative poses which leads to a nearly diagonal system of equations. The alternative of using a minimal independent set of poses to represent the map would require, essentially, that the constraints be substituted into the objective. Doing so, leads to a fully populated coefficient matrix. When the constraints on the poses are represented in a second set of equations as hard constraints, the resulting structure can be solved much more efficiently.

Network Cycle Basis

In the general case, the generation of independent loop equations is itself a difficult problem. The main concern is that the loop equations must be independent to preserve the rank of the coefficient matrix CC^T. In figure 2 for example, although there are nine distinct loops, four of them can be eliminated because there is no node in the center of the figure. The remaining five loops include those formed by each set of three nodes, and the one loop composed of four nodes. Of these five loops, only three are independent.

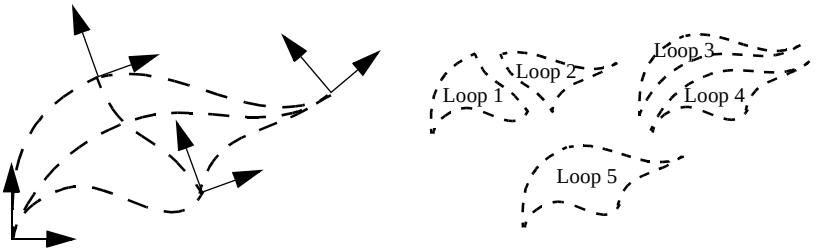


Figure 2: Network Cycle Basis. There are four nodes in this graph and 6 edges. Notice that there is no node at the center of the figure so the edge intersection there is of no consequence. There are five distinct loops but only three of them are independent.

Generating the loop equations requires an algorithm for computing the cycle basis of the network. While the literature on this aspect of network theory devotes significant effort to finding the best basis for a given purpose [20], our results do not depend on the exact basis chosen.

Spanning Tree

In general, the problem of finding a cycle basis is related to the problem of finding a spanning tree of the network. Each different possible spanning tree corresponds to a different cycle basis. In a spanning tree, each node (except an arbitrarily chosen root node) is assigned one and only one parent. Once the tree is constructed, two important statements become true:

- There is now a unique acyclic path between any two nodes in the tree.
- Any edge in the network, but not in the tree, closes an independent loop which is a member of the cycle basis encoded by the tree.

Figure 3 illustrates these matter on a simple tree. The choice of root node and the order of traversal determines the form of the tree. Notice that if there are N nodes, then the spanning tree contains $N-1$ edges. If there are E edges in the original network, then there must be:

$$L = E-(N-1) = E-N+1 \qquad (9)$$

independent loops.

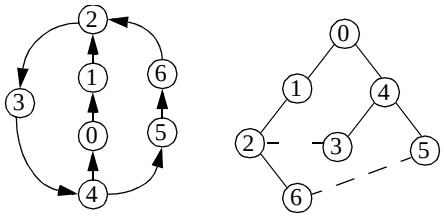


Figure 3: Spanning Tree and Cycle Basis. The network of 7 nodes has two independent loops. The spanning tree generated when node 0 is chosen as the root is illustrated. The edges between nodes (5,6) and (2,3) are the identified loop closing edges.

The spanning tree is generated by a breadth first traversal of the pose network. For this purpose, edges are considered bidirectional. Starting from the root, each node and edge is marked as it is traversed. When any node is encountered for the second time:

- The edge traversed is marked as a loop closing edge and not included in the spanning tree.
- The node is not elaborated into its children.

When there are no nodes left in the traversal queue, the tree is complete. This algorithm is clearly of complexity $N + E$ because each edge and node is visited only once.

The generation of loop constraint equations requires the generation of an explicit sequence of edges which comprise each loop in the cycle basis. A sequence of loop edges is comprised of the path in the spanning tree and the edge outside the tree which closes the loop. It is important to track whether the edges in the loop are traversed in the same sense or the opposite sense of the relative pose to which they correspond.

A path between any two nodes, through the root, is implicitly encoded in the parent links of the spanning tree. However, for generating loop equations, it is necessary to detect if two such paths to the root converge before they reach the root. Detecting this case is straightforward if the depth of a node is computed during spanning tree creation and the paths to the root are elaborated alternately one step at a time after the path from the deeper node is traversed up to the depth of the shallower.

The algorithm used to construct the path is:

- Start at the deeper of the two nodes and follow its parent links until the depth of the shallower node is reached.
- While the last parent node encountered in both paths is not the same node, move both paths toward the root by one step.
- The loop sequence is then the path from the deeper node to the common node, the reverse of the path from the shallow node to the common node, and the loop closing edge.

This algorithm has a complexity bounded from above by the length of the largest loop times the number of loops. Therefore, in no case can it exceed LE operations.

Results

The constrained optimization mapping system has been recently implemented and is undergoing field trials. Presently, images are captured roughly ten times a second. As a result, there is little difference in the quality of maps produced using the full constrained optimization approach versus simply enforcing the constraints without optimization. The latter amounts to a right pseudoinverse applied to all loops simultaneously. Consecutive scans are registered using the iterative closest point (ICP) algorithm before the mapping process commences. Computing times required to register consecutive scans in this manner are not included in the times provided because these calculations can be performed on-line while the images are being captured. The processor used is a 1.5 GHz Pentium with 1 Gbyte of RAM. The two example trials provided below have been chosen to exemplify how very large maps can be rendered globally consistent in comparatively little computation.

Large Cyclic Facility

One trial involved mapping several floors of Pittsburgh's ice hockey stadium. A map of one floor is presented below both before and after rendering it consistent.

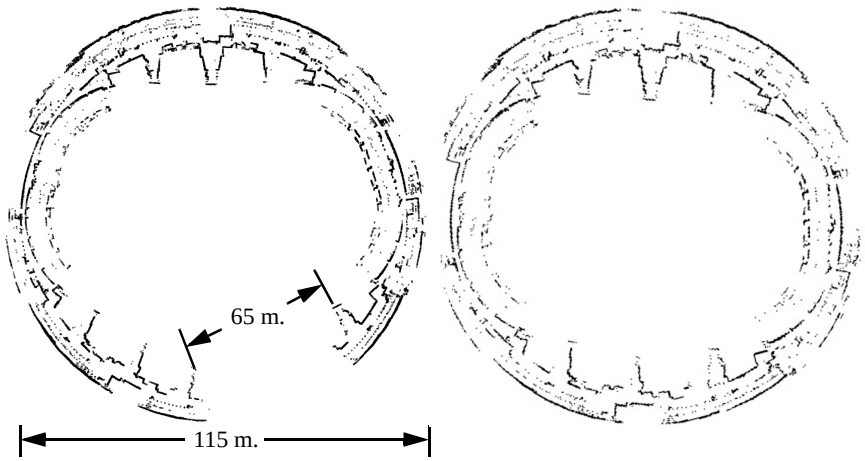


Figure 4: Ladar Map of Pittsburgh's Mellon Arena. 3875 ladar scans have been rendered globally consistent in 0.14 seconds of computation. Left: original odometry-based map. Right: final map.

In this case, four iterations were required to reduce the original 65 meter error at loop closure to under 1 cm. Overall runtime was 0.14 seconds to just enforce constraints and 0.65 seconds to also compute the optimum map. Clearly, a slight curvature error to the right seems evident in the odometry.

Large Multi-Cyclic Facility

The odometry was very well calibrated in this second test. Here, the environment is a typical grocery store. Such environments are ideal for demonstrating the benefits of our approach. There are a large but finite number of loops and the number of images required to map the facility is enough to overwhelm alternative approaches. The vehicle trajectory was a left to right scan of the facility where each set of shelves was encircled once.

The left figure testifies to the excellent odometry calibration in effect for this test. After 11 complete turns and roughly 700 meters of travel, the heading error remains under 20 degrees. This error does, however, account for the gradual skew in the aisles as the vehicle moved from left to right in the figure. Misregistration at the ends of the aisles is evident when they are seen a second time after looping around each set of shelves. Various fuzzy clouds of points in the aisles are either displays or occasional moving people.

The quality of the odometry led to an average misregistration on the order of 3 meters at the point of closure of each of the 23 loops used. Four iterations were required to enforce all constraints to under a cm in 12.22 seconds of computation. For problems on this scale, the extra effort to compute an optimal map is insignificant because the constraint equations require most of the processing.

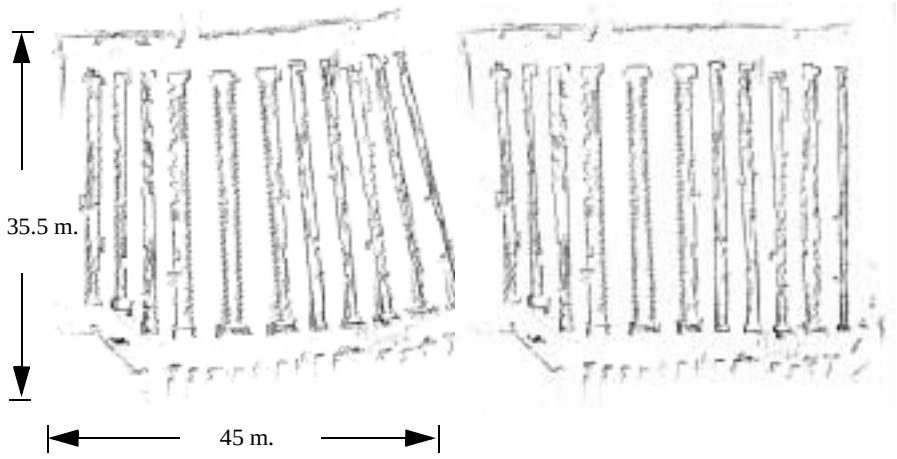


Figure 5: Ladar Map of 12 Aisles of a Grocery Store. 10,380 ladar scans have been rendered globally consistent in 12.22 seconds of computation. Left: original odometry-based map. Right: final map.

Conclusions

This paper has shown the promise of explicit topological reasoning to solve the problem of globally consistent optimal mapping on the large scale. When the pose network describing observations is sparse, as it often is, and the problem size is large, the computational advantages in terms of both memory and processing can be both spectacular and enabling.

The technique is applied here to ladar scans and we have applied it earlier to image mosaicking but it applies in principle to any form of imagery that can be registered whether it is 2D or 3D. Indeed, the fundamental result applies to all constrained optimization problems, and it is likely to have been exploited in other fields for some time.

At an abstract level, sparse cyclic systems are systems which are only slightly overconstrained. For such systems, the process of extracting an explicit expression of the hard constraints encoding "consistency" seems to have great practical value. It may well be that a factorization approach to solving the normal equations can be found which is equivalent but the main point is that either process will require topological reasoning. Very hard sparse optimization problems then become reducible to easy problems in graph theory followed by easy ones in optimization.

While we have demonstrated the use of the technique in a batch processing mode, nothing in the formulation prevents its use for incremental map building. Indeed, its linear complexity implies that optimal consistent maps of large scale facilities could be constructed on the fly as long as the cycle basis can be maintained and a robust automatic mechanism for detecting loop closure can be developed. The latter has been shown by several other researchers including [1] and the former can be accomplished by simply re-running the present algorithm as new loops are closed.

To the degree that limited sensor range and environmental occlusion makes all problems sparse on the large scale, we have begun to support the case that optimal mapping can be accomplished in a fully automatic fashion on such very large scales without requiring any additional assumptions in order to reduce computational complexity.

References

1. J. S. Gutmann and K. Konolige, "Incremental Mapping of Large Cyclic Environments", *Proc. Computational Intelligence in Robots and Automation (CIRA) 1999*, pp. 318-325.
2. F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping", *Autonomous Robots*, 4:pp.333-349, 1997.
3. P. Newman, "On the Structure and Solution of the Simultaneous Localization and Map Building Problem", *Ph.D. thesis submitted in March 1999, Australian Centre for Field Robotics, Univ. of Sydney.*
4. H.S. Sawhney, S. Hsu and R. Kumar, "Robust video mosaicing through topology infer-

ence and local-to-global alignment",*Proc. European Conf. on Computer Vision (ECCV), Freiburg, Germany,* vol.2, pp.103-119, June 1998.

5. R. Unnikrishnan and A. Kelly, "A Constrained Optimization Approach to Globally Consistent Mapping", *Proc. International Conference on Robotics and Systems (IROS) 2002*.
6. J. Guivant and E. Nebot, "Optimization of the Simultaneous Localization and Map Building Algorithm for Real Time Implementation", *IEEE Trans. on Robotics and Automation, June 2001*, 17(3), pp. 242-257
7. P.F. McLauchlan, "The Variable State Dimension Filter applied to Surface-based Structure from Motion", *CVSSP Technical Report VSSP-TR-4/99*
8. Z. Zhang: "Iterative Point Matching for Registration of Free-Form Curves and Surfaces," International Journal of Computer Vision, 1994.
9. P. Besl and N. McKay: "A Method of Registration of 3-D Shapes," IEEE Trans. PAMI 12, 1992.
10. Bergevin, Soucy, Gagnon, and Laurendeau: "Towards a General Multi-View Registration Technique," IEEE Trans. PAMI 18, 1996.
11. P. Motalier, R. Chatila, "Stochastic multisensory data fusion for mobile robot location and environmental modelling", in 5th Symposium on Robotics REsearch, Tokyo, 1989.
12. R. Smith, M. Self, and P. Cheeseman, "Estimating Uncertain Spatial Relationships in Robotics", in Autonomous Robot Vehicles, Springer Verlag, New York, 199). I. Cox, and G Wilfong, editors.
13. J. Guivant and E. Nebot. Optimization of the simultaneous localization and map building algorithm for real time implementation. IEEE Transactions on Robotic and Automation, 17(3):242{257, June 2001.
14. J. A. Castellanos, J. D. Tards, and G. Schmidt, "Building a Global Map of the Environment of a Mobile Robot: The Importance of Correlations.", In Proc. IEEE Conference on Robotics and Automation, pages 1053-1059, 1997.
15. J. Leonard, and H. Feder. "A Computationally Efficient Method for Large-Scale Concurrent Mapping and Localization", in Proc 9th International Symposium on Robotics Research.
16. M. Thompson, R. Eller, W. Radlinski, and J. Speert, editors, *Manual of Photogrammetry,* American Society of Photogrammertry, 3rd Edition, 1966.
17. H.F. Durrant-Whyte, "Integration, Coordination, and Control of Multi-Sensor Robot Systems", esp. chapter 2, Kluwer, 1988.
18. Chartrand, "Graphs as Mathematical Models", Pringle 1977.
19. Blais, G., and Levine, M., "Registering Multiview Range Data to Create 3D Computer Objects", Trans PMAI, Vol 17, No. 8, 1995.
20. Deo N., Prabhu G.M. and Krishnamoorthy M.S., *Algorithms for generating fundamental cycles in a graph*, ACM Trans. Math. Software, 8 (1982), pp. 26-42.