

Helmert’s and Bowie’s Geodetic Mapping Methods and Their Relation to Graph-Based SLAM

Pratik Agarwal

Wolfram Burgard

Cyrill Stachniss

Abstract—The problem of simultaneously localizing a robot and modeling the environment is a prerequisite for several robotic applications and a large variety of solutions have been proposed allowing robots to build maps and use them for navigation. Also the geodetic community addressed large-scale mapping for centuries, computing maps which span across continents. These mapping processes had to deal with several challenges that are similar to those of the robotics community. In this paper, we explain two key geodetic mapping methods that we believe are relevant for robotics. We also aim at providing a geodetic perspective on current state-of-the-art SLAM methods and at identifying similarities between the solutions proposed by both communities. The central goal of this paper is to bring both fields close together and to enable future synergies.

I. INTRODUCTION

The problem of simultaneously localizing and mapping (SLAM) is essential for most robotic applications. Most modern SLAM methods follow the graph-based SLAM paradigm [29], [16], [37], [20] in which each pose of the robot or each landmark position is represented as a node in a graph. A constraint between two nodes, which results from observations or odometry, is represented by an edge in the graph. The first part of the overall problem is to create the graph and is often referred to as the front-end. The second part deals with finding the configuration of the nodes that best explains the constraints modeled by the edges and is typically referred to as a back-end.

In the geodetic mapping community, one major goal has been to build massive survey maps, some even spanning across continents. These maps can then be used either directly by humans or for studying large scale properties of the earth. In principle, geodetic maps are built in a similar way to the front-end/back-end approach used in SLAM. Constraints are acquired through observations between physical observation towers. These towers correspond to positions of the robot as well as the landmarks in the context of SLAM. Once the constraints between observation towers are obtained, the goal is to optimize the resulting system of equations to get the best possible locations of the towers on the surface of the earth. Figure 1 illustrates two such large scale geodetic network of constraints used for mapping Europe and North America.

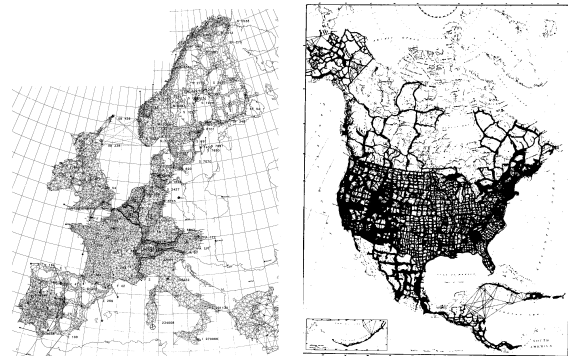


Fig. 1. Left: Triangulation network spanning across Europe for ED 87 (courtesy of [14]). Right: North American network for NAD 87 (courtesy of [35]). The NAD 87 network contained roughly 270,000 stations and required solving 1.8 million equations [8].

SLAM and geodetic mapping have problems in common. The first challenge is the large size of the maps, i.e., the number of unknowns. In geodetic mapping, the unknowns are the positions of the observation towers, while for robotics, the unknowns corresponds to robot positions and observed landmarks. For example, the system of equations for the North American Datum of 1927 (NAD 27) required solving for 25,000 tower positions and the North American Datum of 1983 (NAD 83) required solving for 270,000 positions [26].

In robotics, front-ends may be unable to distinguish between similar looking places, which leads to perceptual aliasing. In geodetic mapping, the front-end consisted of humans carefully and meticulously acquiring measurements. Even this process was prone to making mistakes [35]. Another challenge comes from the non-linearity of constraints, which is frequently the case in SLAM as well as geodetic mapping. A commonly used approximation in both fields is to linearize the problem around an initial guess.

These similarities warrant studying geodetic mapping techniques and we believe it may inspire novel solutions to large-scale, autonomous robotic SLAM. The aim of this paper is to survey two key approaches of the geodetic mapping community and put them in relation to recent SLAM research. This mainly relates to the Bowie method [9], [1] used for creating NAD 27 and the Helmert blocking method [23] used in optimizing the set of equations for NAD 83.

In the remainder of this paper, we first provide a brief review of graph-based SLAM. Second, we explain the problem formulation for geodetic mapping and introduce commonly used terminologies. We continue to explain the approaches

All authors are with the University of Freiburg, Institute of Computer Science, 79110 Freiburg, Germany. Cyrill Stachniss is also with the University of Bonn, Inst. of Geodesy and Geoinformation, 53115 Bonn, Germany. This work has been partially supported by BMBF under contract number 13EZ1129B-iView, by the MWK for the project ZAFL-AAL and by the EC under contract numbers ERG-AG-PE7-267686-LifeNav and FP7-600890-ROVINA.

to geodetic mapping including the motivation and insights central to Bowie’s and Helmert’s solutions. We then highlight the relationships between existing SLAM approaches and these geodetic mapping techniques. An extended survey on geodetic approaches covering other topics such as variable reordering, robustness to outliers, and additional optimization methods can be found in [3].

II. GRAPH-BASED SLAM IN ROBOTICS

Graph-Based SLAM methods model the individual poses of a robot as well as landmarks as nodes in a graph. These nodes are connected through constraints whereas each constraints can result from odometry, inertial measurement units, sensor measurements, or similar. Graph-Based SLAM back-ends aim at finding the configuration of the nodes that minimize the error induced by constraints. Let $\mathbf{x} = (x_1, \dots, x_n)^T$ be the state vector where x_i describes the pose of node i . This pose x_i is typically three-dimensional for a robot living in the 2D plane. We can describe the error function $\mathbf{e}_{ij}(\mathbf{x})$ for a single constraint between the nodes i and j as the difference between the obtained measurement z_{ij} and the expected measurement $f(x_i, x_j)$ given the current state

$$\mathbf{e}_{ij}(\mathbf{x}) = f(x_i, x_j) - z_{ij}. \quad (1)$$

The realization of the measurement function f depends on the sensor setup. For pose to pose constraints, one typically uses the transformation between the poses. For pose to landmark constraints, we minimize the reprojection error of the observed landmark into the frame of the observing pose. The error minimization can be written as

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmin}} \sum_{ij} \mathbf{e}_{ij}(\mathbf{x})^T \Omega_{ij} \mathbf{e}_{ij}(\mathbf{x}), \quad (2)$$

where Ω_{ij} is the information matrix associated to a constraint and \mathbf{x}^* is the optimal configuration of nodes with minimum sum of error induced by the edges. Linearizing the error function leads to a quadratic form that can be minimized by computing its derivative and setting up a linear system.

Multiple improvements have been proposed. First, it is common to solve the linear system using domain knowledge and sparse linear algebra methods. Folkesson and Christensen formulate the least squares problem as an energy-minimization problem [16]. Dellaert and Kaess explore the graph SLAM formulation as a factor graph using smoothing and mapping [12] and use matrix factorization methods such as QR, LU, and Cholesky decomposition. Other authors suggest to perform the minimization using relaxation techniques [24], [13], [17], [34], stochastic gradient descent [32] or variants of that [22].

Second, for dealing with outliers in the constraints, different methods have been proposed [5], [33], [36]. Third, performing the operations in a non-Euclidean manifold space allow for better dealing with the angular components. Finally, hierarchical and submapping approaches have shown to efficiently compute solutions by decomposing the optimization problem at different levels of granularity [7], [15], [21], [31].

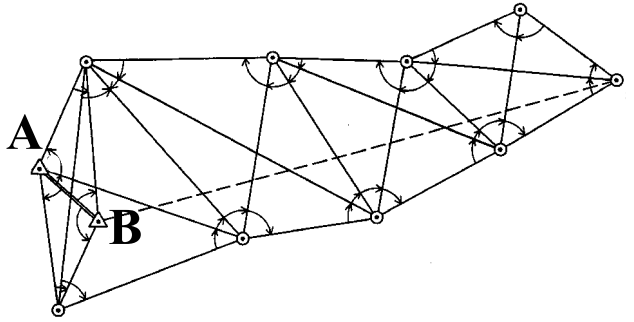


Fig. 2. A simple triangle net. The distance between towers A and B is physically measured. Angular constraints to all other towers are measured and the lengths of all other segments are computed with respect to the baseline AB with the measured angles. Courtesy of [10].

III. GEODETIC MAPPING

Geodetics, also known as geodesy, is the science that studies the earth at various global and local scales. It measures, for example, large scale changes in the earth’s crustal movements, tidal waves, magnetic or gravitational fields. The aspects of geodesy that are relevant for addressing the SLAM problem, are related to geodetic mapping. In this section, we first describe how the massive geodetic surveys, which typically result in *triangle nets*, are created. We then explain the Bowie and Helmert [23] method for obtaining a minimal error configuration.

The basic principle behind geodetic surveying is triangulation. A large set of observation towers, called Bibly towers and typically 20m to 30m in height, are built from which other towers are observed [10]. Geodetic surveyors built large interconnected mesh of observation towers by triangulating measurements between them. The resulting mesh of towers and constraints is commonly called a triangle net. A simple example of a triangle net is shown in Figure 2. Each line segment in Figure 2 is a constraint between two observation towers. Some of these constraints are directly measured while others are computed using trigonometrical relationships.

The method of obtaining constraints between the observing towers has evolved over time. Initially, constraints were distance only or angle only measurements. The distance measurements were obtained using tapes and wires of Invar, which has a low coefficient of thermal expansion. Later, more sophisticated instruments, such as parallax range-finders, Stadimeter, Tellurometer, and Electronic Distance Measuring Instruments were used. Angular measurements were obtained using theodolites. Moreover, measurement constraints used in geodetic surveys can be differentiated as absolute and relative measurements. Absolute measurements involve directly measuring the position of a tower on the surface of the earth. These include measuring latitudes by observing stars at precise times and then comparing them to the existing catalogs. Stars were also used as fixed landmarks for bearing only measurements from different base towers at precisely the same time. More sophisticated techniques such as space systems like very long baseline interferometry position differences and GPS measurements were later introduced

with improvements in measurement techniques. Once GPS technology became accurate, only GPS measurements were used to locate absolute positions of each station. In the United States the new set of observations containing only GPS came to be known as Continuously Operating Reference Stations (CORS). Currently new versions of CORS are released by National Oceanic and Atmospheric Administration (NOAA).

Examples of large scale triangle nets are shown in Figure 1. The left image displays the geodetic triangle net used for mapping Central Europe in 1987 whereas the right one shows the triangulation network for mapping North-America in 1983 (NAD 83). The thick lines in NAD 83 are long sections of triangulation nets comprising of thousands of Bibby towers. Connections between multiple sections are small triangle networks and are called junctions.

The geodetic mapping problem can be broken down into two major sub problems, the first being the “adjustment of the vertical”, the second being “adjustment of the net”. The problem of “adjustment of the net” is finding the least squares system of the planar triangulation net whereas “adjustment of the vertical” involves finding parameters to wrap this mesh network on a geoid representing earth [39].

IV. ADJUSTMENT OF THE NET – GEODETIC BACK-ENDS

The problem of adjustment of the net is similar to the graph-based SLAM formulation. In the SLAM notation, the geodetic mesh network consists of observation towers constrained by non-linear measurements. These physical towers are similar to the unknown state vector $\mathbf{x} = (x_1, \dots, x_n)^T$ in the SLAM problem.

The task of the back-end optimizer in geodesy is to find the best possible configuration of the towers on the surface of the earth to minimize the sum of errors from all constraints. All non-linear constraints can be linearized using Taylor series expansion and stacked into a matrix [26]. The least squares solution can be computed directly by solving the least square system. Both, SLAM and geodetic problems inherently contain non-linear equations constraining a set of positional nodes. The process of linearizing the constraints and solving the linearized system is repeated to improve the solution.

The biggest challenge faced by the geodetic community over centuries was limited computing power. Even with the most efficient storage and sparse matrix techniques, there was no computer available which could solve the full system of equations as a single problem. Bowie proposed the first feasible approximation of solving the full system of equations for creating NAD 27.

A. The Bowie Method

The network of constraints during early geodetic mapping were built in sections and junctions. Sections are long chain like traverses which join other sections in junctions. Bowie’s main insight was that he can approximate the net comprising of sections by collapsing intersections into a single node and the sections into a single virtual constraint. This is illustrated in Figure 3. This much smaller least squares system is solved

to recover the positions of the intersections, which can then be used to compute the sections independently.

The core steps of the Bowie method consist of separating the sets of unknowns into segments and junctions. The junctions act as a small set of separator nodes. The junction nodes are shown as circles in Figure 3. These are not a single node but a small subnet of towers, which separates the sections. All nodes in one junction are optimized together using least squares adjustment but ignore any inter-junction measurements. After this optimization, the structure of the junction does not change, i.e., each node in a junction is fixed with respect to other nodes in that particular junction.

As a next step, new latitudinal and longitudinal constraints are created between junctions. This is done by approximating each section with a *single* constraint. Each of such single constraints is a two-dimensional longitude and latitude constraint. As a result, each junction turns into a single node and each section into a single constraint. This leads to a much smaller but approximate problem, which is optimized using the full least-squares approach.

The above described steps of the Bowie method are summarized in Alg. 1, see also Adams [1]. The full least squares problem is not solved by matrix inversion but by a variant of Gaussian elimination called Doolittle decomposition (see Wolf [40]). The Doolittle algorithm is a LU matrix decomposition method, which decomposes a matrix column-wise into a lower triangular and upper triangular matrix. The solution can be computed using forward and backward substitution steps as with other matrix decomposition methods as well.

Algorithm 1 Bowie method

- 1: Separate triangle net into junctions and segments
 - 2: Optimize each junction separately
 - 3: Create new virtual equations between junctions treated as a single node
 - 4: Solve the abstract system of equations comprising of each junction as a single node and each section as a single constraint
 - 5: Update the resulting positions of stations in the segments using the new junction values
-

In essence, the Bowie method generates new, virtual constraints from sections. The uncertainty of such a virtual constraint is not equal to the uncertainty of an individual measurement in the section. Thus, Bowie introduces a weight for each virtual measurement. These weights are chosen as the ratio of the length of the section with respect to the sum of all the section lengths. Hence, the weights are proportional to the length of the sections so that the larger proportion of the error is distributed over long sections compared to shorter ones.

Another computational trick to lower the efforts is to use diagonal covariances for the two-dimensional, virtual, latitude/longitude constraints. This enables to separate the system of equations for longitude and latitude. This yields two least squares problems with half of the original size.

The partitioning of the triangular net into junctions and sections is done manually. Figure 3 illustrates the original triangle net and Bowie’s approximated net into segments

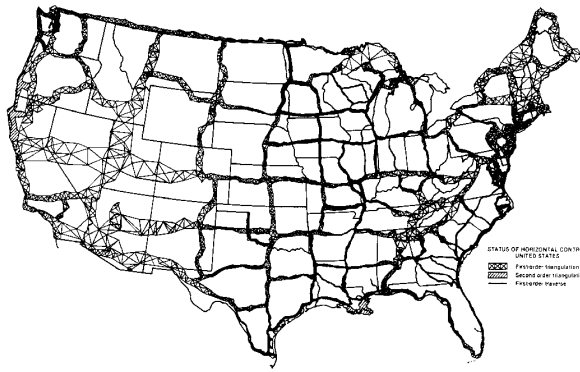


Fig. 3. Bowie method as used for NAD 27 triangle net (left) and the resulting approximation (right). Each small circle represent a junction and the junctions are connected by sections. Given the values of the junctions, the segments are independent of the rest. Courtesy of [35].

and junctions used for NAD 27. In Figure 3, the small circles represent junctions and all lines connecting junctions are sections of triangle nets. These sections and junctions individually represent a subset of the constraints connecting stations.

To the best of our knowledge, the Bowie method is the first implementation of a large scale approximate least squares system. It was effectively used in creating the NAD 27 as the triangulation nets were build in sections forming large loops. The Bowie methods exploits this structure. With time, more triangulation nets have been created yielding a large number of new loops. In this process, initially new triangulation nets were not optimized as a whole. They were integrated into the existing system by keeping the previous positions fixed. As the network of constraints became even more denser a complete optimization of the network of constraints was required. Helmert’s blocking method was chosen for this new optimization process to optimize the constraints tractably for creating NAD 83.

B. Helmert Blocking

Helmert proposed the first method for solving the large optimization problem arising from the triangulation network in parallel. His strategy, which was proposed in 1880, is possibly the oldest technique to partition the set of equations into smaller problems [23], [2]. Helmert observed that by partitioning the triangle net in a particular way, one can solve the overall system of equations in parallel. He outlined that the whole triangle net can be broken into multiple smaller subnets. All nodes that have constraints only within the same subnet are called inner nodes and can be eliminated. All separator-nodes, i.e., those that connect multiple subnets, are then optimized. The previously-eliminated inner nodes can be computed independently given the values of the separator nodes. As a result, the so-formed subnets can be solved in parallel. Helmert’s blocking method is outlined in Alg. 2 and is explained more precisely as a mathematical algorithm by Wolf in [41].

Consider a simple triangle net shown in Figure 4. In Figure 4(a), each line segment is a constraint and the end

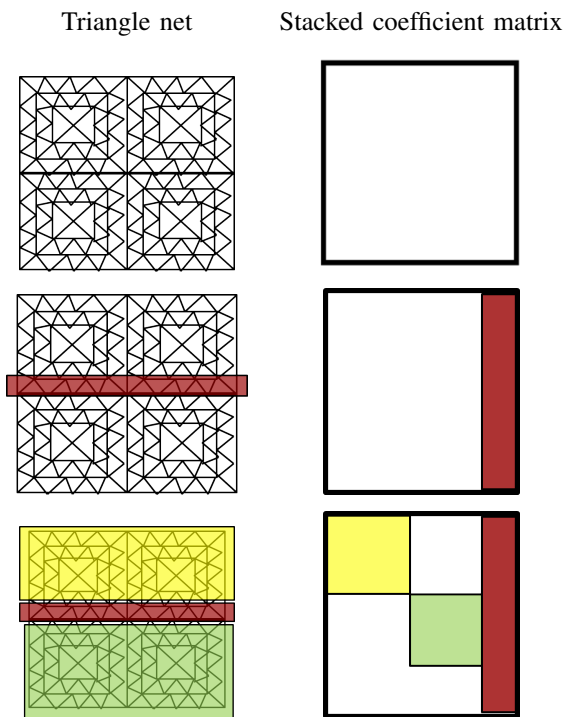


Fig. 4. Helmert blocking in action. The left column shows a toy example of triangle net. The right column shows the corresponding stacked coefficient matrix for each net.

of segments represent a physical observation tower. Helmert observed that if he divides the triangle net into two halves, for example as illustrated in Figure 4(b), the top half of the towers will be independent of the bottom half given the values of the separators as shown in Figure 4(c). Such a system can be solved using reduced normal equations [41], [35].

Let us represent the whole system of equations from the triangle net in Figure 4(a), as:

$$Ax = b. \quad (3)$$

This equation can be subdivided into 3 parts in the following manner:

$$\begin{bmatrix} A_s & A_1 & A_2 \end{bmatrix} \begin{bmatrix} x_s \\ x_1 \\ x_2 \end{bmatrix} = b. \quad (4)$$

Here, A_s and x_s represent the coefficients and unknowns, arising from the central separator. A_1 and A_2 are coefficients of the top and bottom subnets. The coefficient matrix $\begin{bmatrix} A_s & A_1 & A_2 \end{bmatrix}$ in Eq. 4 is shown on the right-hand side of Figure 4(c). The corresponding system of normal equations is:

$$\begin{bmatrix} N_s & N_1 & N_2 \\ N_1^T & N_{11} & 0 \\ N_2^T & 0 & N_{22} \end{bmatrix} \begin{bmatrix} x_s \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_s \\ b_1 \\ b_2 \end{bmatrix}. \quad (5)$$

The towers in x_1 do not share any constraints with towers in x_2 . Both, x_1 and x_2 share constraints with x_s but not with each other. The key element in Eq. 5 is the block structure of N_{11} and N_{22} . The system of equation in Eq. 5 can be reduced such that:

$$\bar{N}_s x_s = \bar{b}_s, \quad (6)$$

where \bar{N}_s is computed as:

$$\begin{aligned} \bar{N}_s &= N_s - \begin{bmatrix} N_1 & N_2 \end{bmatrix} \begin{bmatrix} N_{11}^{-1} & 0 \\ 0 & N_{22}^{-1} \end{bmatrix} \begin{bmatrix} N_1^T \\ N_2^T \end{bmatrix} \\ &= N_s - \sum_{i=1,2} N_i N_{ii}^{-1} N_i^T, \end{aligned} \quad (7)$$

and \bar{b}_s is computed as:

$$\bar{b}_s = b_s - \sum_{i=1,2} N_i N_{ii}^{-1} b_i^T. \quad (8)$$

Once \bar{N}_s has been solved, x_1 and x_2 can be computed by solving:

$$N_{11} x_1 = b_1 - N_s^T x_s \quad (9)$$

$$N_{22} x_2 = b_2 - N_s^T x_s. \quad (10)$$

Moreover, Wolf states that matrices should never be inverted for computing Eq. 6, 9, and 10 [41]. Instead, Cholesky decomposition or Doolittle-based LU decomposition should be employed. The steps outlined for computing the reduced normal forms in Eq. 7 and 8 represent the Schur complement. The inverse computation in this step is trivial if the subnets result in block diagonal matrices. If not, both Wolf and Schwarz mention that each of the subnets can themselves be sparse and can be further sub divided recursively [41], [35].

Algorithm 2 Helmert Blocking

- 1: Given a partitioning scheme, establish the normal equations for each partial subnet separately
- 2: Eliminate the unknowns for all nodes which do not have constraints with neighboring partial subnets
- 3: Established the main system after eliminating all the inner nodes containing only intra-subnet measurements
- 4: Solve the main system of equations containing only separator nodes
- 5: Solve for inner nodes given the value of separator nodes

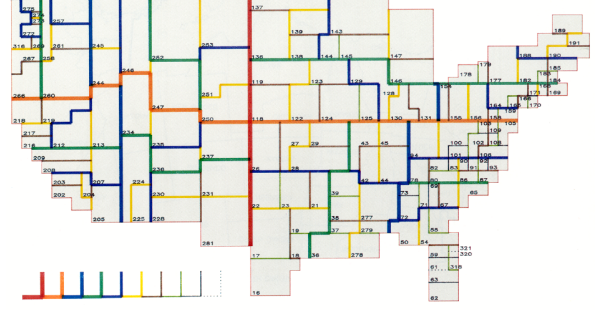


Fig. 5. Helmert blocking applied to North America with ten levels. The legend in the bottom left corner shows progressive levels of partitions. The first partition, cuts the continent into east-west blocks, whereas the second cut partitions each half into north-south blocks. Hence, each geographical block is partitioned into four regions and this method is recursively performed on each sub-block. Courtesy of [35].

Helmert himself provided simple instructions for creating subnets and for partitioning the blocks, which is critical for his method. He first instructed to pick a latitude such that it partitions all the towers roughly into halves. Next, a longitude is chosen for each upper and lower half to partition the Northern and Southern regions into Eastern and Western partitions. Each obtained geographical rectangular block is recursively partitioned into four further blocks. The proposed method is simple but effective, since the Geodetic triangulation network is built roughly as a planar graph and the density of the net was approximately similar across different locations. Helmert's approach to partitioning triangle nets also shares resemblance to nested dissection variable reordering strategies [18] used to efficiently factorize sparse matrices. The use of variable re-ordering significantly improves the computation and memory requirements for matrix decomposition methods [11], [4].

The nested dissection variable reordering scheme was initially proposed for solving a system of equations for $n \times n$ grid problems arising from finite-element discretization of a mesh [18]. Helmert's proposal to divide a triangulation net recursively along latitudes and longitudes has strong resemblance with the nested dissection variable reordering scheme to minimize fill-in in matrix decomposition methods [6]. It was used by Avila and Tomlin for solving large least squares using the ILLIAC IV parallel processor for optimizing geodetic networks [6]. Later, Golub and Plemmons used the Helmert blocking variable ordering strategy for solving large system of equations using orthogonal decomposition techniques such as QR decomposition for the system of equations arising from the geodetic network [19].

In sum, the Bowie method is an approximation of Helmert Blocking. The approximation uses single level subnets and an approximate optimization of the junction nodes. The optimization of the highest level in Helmert Blocking consists of junction nodes and is computed via reduced normal equations. In contrast to that, the Bowie method uses virtual constraints at the highest level. The main difference is that the system of equations created by the virtual constraints

is much smaller and sparser and hence easier to optimize than the full set of reduced normals as in the exact Helmert blocking method.

Although graph partitioning and submapping has been frequently used in robotics, to the best of our knowledge, Helmert’s method has not been referenced in the robot mapping community—only Triggs et al. [38] mention it as an optimization procedure in their bundle adjustment survey.

V. RELATION TO SLAM BACK-ENDS

The SLAM community started using sparse linear algebra and matrix decomposition methods recently [12], [28], [25]. In contrast to that, the geodetic scholars were using LU and Cholesky decomposition for a long time—Cholesky decomposition was developed in the early 1900s for geodesy and map building. This section aims at highlighting some of the similarities between the methods developed by both communities.

Grisetti et al. [21] propose an efficient hierarchical multi-level method for optimizing pose-graphs. The higher the level in the hierarchy, the smaller the pose-graph. At each level, a subgraph from a lower level is represented by a single node on the higher level. The optimization is primarily carried out at the higher levels and only propagated down to the lower level if there is a significant update to the pose of a node. Each edge in the higher level is computed via a new deduced virtual measurement, which is created after a local optimization. The gain in speed by this method results from the fact that computationally expensive optimizations are only performed when the pose of a node in the coarse representation gets affected more than a certain threshold.

The virtual measurements created by Grisetti et al. are from the idea itself similar to those in the Bowie method explained in Section IV-A. The Bowie method creates a two-level hierarchy instead of the multiple levels as in [21]. Both methods use a single node from a dense sub-graph in the higher, coarser level and add a virtual deduced measurement between the smaller new problem instances. A difference is that given a geodetic network consisting of sections, each edge in the Bowie method represents a sub-graph whereas in the hierarchical approach of Grisetti et al., only nodes represent sub-graphs. Furthermore, Grisetti et al. compute the uncertainty of a virtual constraint explicitly.

Also Ni et al. propose back-ends, which divide the original problem into multiple smaller sub problems [31], [30]. These smaller problems are then optimized in parallel. The first approach [31], partitions the problem into a single level of sub-maps, while [30] partitions each map recursively multiple times. The speed-up is mainly due to caching the linearization result of each subgraph. In both methods, all nodes in a subgraph are expressed with respect to a single base node. This allows for efficiently re-using the linearization of the constraints within each subgraph. This insight results in a reduction of the total computation time. Boundary nodes and constraints, which connect multiple subgraphs, need to be recomputed at each iteration, while cached results for constraints concerning nodes inside a subgraph can be

reused. The methods of Ni et al. would result in a batch solution if each non-linear constraint within a subgraph is re-linearized at every iteration, but they show experimentally that by not doing so, a high speed-up in optimization time is achieved at the expense of small errors.

The sub-maps methods proposed by Ni et al. show strong similarities to Helmert’s approach of partitioning the triangle nets into subnets. The idea of anchoring sub-maps in [30] has a similar motivation as Bowie’s idea for anchoring junctions and moving them as a whole by shifting the anchor nodes and not re-optimizing each subgraph. The recursive nested-dissection algorithm, which is at the heart of Ni et al.’s sub-map-based methods, is similar to Helmert’s strategy of recursively partitioning a planar triangulation net [19]. Furthermore, the out-of-core parallel Cholesky decomposition-based non-linear solver for geodetic mapping proposed by Avila and Tomlin in 1979 [6] uses Helmert blocking and is thus strongly connected to [30]. Note, however, that the geodetic community was generating the partitioning manually with domain knowledge and that the triangle-nets have a simpler and more planar structure than typical SLAM graphs.

VI. CHALLENGES IN ROBOTIC MAPPING

All though there are similarities between the problems of both communities, it is also important to highlight the additional challenges that autonomous robots, which rely on working SLAM solutions, face compared to geodetic mapping.

First, SLAM systems are fully automated while geodetic mapping inherently involves humans at all levels of the process. It is difficult for automated front-end data-association methods to distinguish between visually similar but physically different places and this is likely to occur, for example, in large man-made buildings. Perceptual aliasing creates false constraints, which often introduces errors in the localization and map building process. Data association errors in geodetic survey can be assumed to be more local.

Second, the quality of the initial guess is often different. The initial configurations that is available for geodetic triangle networks are typically substantially better than the pose initializations of typical wheeled robots using odometry as well as flying or walking robots. A good initial guess simplifies or even enables the use of polygon filling and other types of approximations.

Third, the geodetic triangle networks are almost planar, while most SLAM graphs are not. Helmert’s simple partitioning scheme of segmenting along latitudes and longitudes works only because the graphs are almost planar. It has been shown that for planar graphs, recursive nested dissection [18] is optimal. In [27], the authors prove that planar or approximately planar SLAM graphs can be optimized in $O(n^{1.5})$ by using the nested dissection reordering. Comparable results can be expected for Helmert’s Blocking strategy given their similarity. The way most modern SLAM methods work, however, leads to a highly non-planar graph.

VII. CONCLUSION

This paper provides a survey of two geodetic mapping methods and aims at providing a geodetic perspective on SLAM. We showed that both fields share similarities when it comes to the error minimization tasks. There are, however, also differences: geodetic triangular nets have a simpler structure, which can be exploited in the optimization, methods for robotics must be autonomous while the geodetic surveys always have humans in the loop, and often the geodetic community had a better initial configuration to begin with. The central motivation for this paper is to connect both fields and to enable future synergies among them.

SLAM researchers have often gone back to the graph theory and sparse linear algebra community for efficient algorithms. It is probably worth to also look into the geodetic mapping literature given that they addressed large-scale error minimization and developed highly innovative solutions to solve them.

ACKNOWLEDGEMENT

We thank Edwin Olson for introducing the work of Schwarz [35] to the authors and we would like to thank our librarian Susanne Hauser for locating some of the old and often difficult to obtain manuscripts. Thanks to Google scholar as a useful resource for finding especially old documents.

REFERENCES

- [1] O.S. Adams. The bowie method of triangulation adjustment, as applied to the first-order net in the Western part of the United States. *US Dep. of commerce, coast and geodetic survey, Special Publ. No. 159*, 1930.
- [2] Aeronautical Chart and Information Center. Preface and Part I: The Mathematical Theories, 1964.
- [3] P. Agarwal, W. Burgard, and C. Stachniss. A survey of geodetic approaches to mapping and the relationship to graph-based slam. *IEEE Robotics and Automation Magazine*. Currently under review.
- [4] P. Agarwal and E. Olson. Variable reordering strategies for SLAM. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2012.
- [5] P. Agarwal, G.D. Tipaldi, L. Spinello, C. Stachniss, and W. Burgard. Robust map optimization using dynamic covariance scaling. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2013.
- [6] J.K. Avila and J.A. Tomlin. Solution of very large least squares problems by nested dissection on a parallel processor. In *Proc. of the 12th Symposium on the Interface*, pages 9–14, 1979.
- [7] M. Bosse, P. Newman, J. Leonard, and S. Teller. Simultaneous localization and map building in large-scale cyclic environments using the Atlas framework. *Int. Journal of Robotics Research*, 23(12):1113–1139, 2004.
- [8] J.D. Bossler. Geodesy solves 900,000 equations simultaneously. *Eos, Transactions American Geophysical Union*, 68(23):569–569, 1987.
- [9] W. Bowie. The triangulation of north america. *The Geographical Journal*, 72(4):348–356, 1928.
- [10] R.K. Burkard. *Geodesy for the Layman*. US Department of commerce, National Oceanic and Atmospheric Administration, 1985.
- [11] T.A. Davis. *Direct methods for sparse linear systems*, volume 2. Society for Industrial and Applied Mathematics, 2006.
- [12] F. Dellaert and M. Kaess. Square root SAM: Simultaneous localization and mapping via square root information smoothing. *Int. Journal of Robotics Research*, 25(12), 2006.
- [13] T. Duckett, S. Marsland, and J. Shapiro. Learning globally consistent maps by relaxation. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, volume 4, pages 3841–3846, 2000.
- [14] W. Ehrnsperger. The ED87 Adjustment. *Journal of Geodesy*, 65(1):28–43, 1991.
- [15] C. Estrada, J. Neira, and J.D. Tardós. Hierarchical SLAM: Real-time accurate mapping of large environments. *IEEE Transactions on Robotics*, 21(4):588–596, 2005.
- [16] J. Folkesson and H. I. Christensen. Graphical SLAM - a self-correcting map. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2004.
- [17] U. Frese, P. Larsson, and T. Duckett. A multilevel relaxation algorithm for simultaneous localization and mapping. *IEEE Transactions on Robotics*, 21(2):196–207, 2005.
- [18] A. George. Nested dissection of a regular finite element mesh. *SIAM Journal on Numerical Analysis*, 10(2):345–363, 1973.
- [19] G.H. Golub and R.J. Plemmons. Large-scale geodetic least-squares adjustment by dissection and orthogonal decomposition. *Linear Algebra and Its Applications*, 34:3–28, 1980.
- [20] G. Grisetti, R. Kümmerle, C. Stachniss, and W. Burgard. A tutorial on graph-based SLAM. *IEEE Transactions on Intelligent Transportation Systems Magazine*, 2, 2010.
- [21] G. Grisetti, R. Kümmerle, C. Stachniss, U. Frese, and C. Hertzberg. Hierarchical optimization on manifolds for online 2D and 3D mapping. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2010.
- [22] G. Grisetti, C. Stachniss, S. Grzonka, and W. Burgard. A tree parameterization for efficiently computing maximum likelihood maps using gradient descent. In *Proc. of Robotics: Science and Systems*, 2007.
- [23] F.R. Helmert. Die mathematischen und physikalischen Theorien der höheren Geodäsie-Einleitung, 1880.
- [24] Andrew Howard, Maja J Mataric, and Gaurav Sukhatme. Relaxation on a mesh: a formalism for generalized localization. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2001.
- [25] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J.J. Leonard, and F. Dellaert. iSAM2: Incremental smoothing and mapping using the Bayes tree. *Int. Journal of Robotics Research*, 31, 2012.
- [26] G.B. Kolata. Geodesy: dealing with an enormous computer task. *Science*, 200(4340):421–466, 1978.
- [27] Peter Krauthausen, Frank Dellaert, and Alexander Kipp. Exploiting locality by nested dissection for square root smoothing and mapping. In *Proc. of Robotics: Science and Systems*, 2006.
- [28] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard. g2o: A general framework for graph optimization. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2011.
- [29] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4, 1997.
- [30] K. Ni and F. Dellaert. Multi-Level submap based SLAM using nested dissection. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2010.
- [31] K. Ni, D. Steedly, and F. Dellaert. Tectonic SAM: Exact; out-of-core; submap-based slam. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2007.
- [32] E. Olson, J. Leonard, and Seth Teller. Fast iterative optimization of pose graphs with poor initial estimates. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2006.
- [33] Edwin Olson and Pratik Agarwal. Inference on networks of mixtures for robust robot mapping. *Int. Journal of Robotics Research*, July 2013.
- [34] L.M. Paz, J. Neira, and J.D. Tardós. Divide and conquer: EKF slam in $O(n)$. *IEEE Transactions on Robotics*, 26(1), 2008.
- [35] C.R. Schwarz. North American datum of 1983. *Rockville, MD: National Geodetic Survey, Charting and Geodetic Services, National Ocean Service*, 1, 1989.
- [36] N. Sünderhauf and P. Protzel. Towards a robust back-end for pose graph slam. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2012.
- [37] S. Thrun and M. Montemerlo. The GraphSLAM algorithm with applications to large-scale mapping of urban structures. *Int. Journal of Robotics Research*, 25(5-6), 2006.
- [38] B. Triggs, P.F. McLauchlan, R.I. Hartley, and A.W. Fitzgibbon. Bundle adjustment: a modern synthesis. In *Vision algorithms: theory and practice*, pages 298–372. Springer, 2000.
- [39] H. Wolf. Triangulation adjustment general discussion and new procedure. *Bulletin Géodésique*, 16(1):87–104, 1950.
- [40] H. Wolf. Gaußscher Algorithmus (Doolittle-Methode) und Boltzsches Entwicklungsverfahren. *Bulletin Géodésique*, 26(1):445–452, 1952.
- [41] H. Wolf. The helmert block method, its origin and development. In *Proc. of the Int. Symp. on Problems Related to the Redefinition of North American Geodetic Networks*, pages 319–326, 1978.