Single-Cluster Spectral Graph Partitioning
for Robotics Applications

Edwin Olson, Matthew Walter, Seth Teller, and John Leonard
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
32 Vassar Street, Cambridge, MA 02139
Email: eolson@mit.edu, mwalter@mit.edu, teller@csail.mit.edu, jleonard@mit.edu

Abstract— We present SCGP, an algorithm for finding a single cluster of well-connected nodes in a graph. The general problem is NP-hard, but our algorithm produces an approximate solution in $O(N^2)$ time by considering the spectral properties of the graph's adjacency matrix. We show how this algorithm can be used to find sets of self-consistent hypotheses while rejecting incorrect hypotheses, a problem that frequently arises in robotics. We present results from a range-only SLAM system, a polynomial time data association algorithm, and a method for parametric line fitting that can outperform RANSAC.

I. INTRODUCTION

Many problems can be cast in the form of finding a set of consistent hypotheses from a larger set of candidate hypotheses. For example, outlier rejection can be posed as finding the set of maximally consistent inlier measurements from the set of all measurements. Other problems, including parameter estimation (e.g. fitting points to a line), and data association can also be cast into this form.

In general, searching for the best subset of hypotheses requires exponential time in the number of candidate hypotheses. This paper presents an algorithm, Single-Cluster Graph Partitioning (SCGP), which casts the problem as a graph partitioning problem and uses spectral analysis to estimate the optimal set in $O(N^2)$ time, where N is the number of hypotheses being considered.

The basic observation motivating SCGP is that correct hypotheses tend to be pairwise consistent with each other, whereas incorrect hypotheses are only randomly consistent. SCGP uses a pairwise similarity metric to find a cluster of hypotheses which is approximately the set with greatest intra-set similarity.

Graph partitioning is a well-studied problem, at least in the context of identifying *two or more* clusters of highly similar nodes. Single-cluster graph partitioning is different; it seeks to find only a *single* cluster. In the problems explored in this paper, the outliers/incorrect hypotheses are not, in general, pairwise similar to each other.

To introduce the idea of single-cluster partitioning, consider the points in Fig. 1. Given a cloud of points, we wish to recover a cluster of "inliers". We compute an adjacency matrix using a simple exponentially-weighted Euclidean distance metric and perform clustering using Shi and Malik's Normalized Cuts [1] (a common k-way partitioning algorithm) and with our 1-way
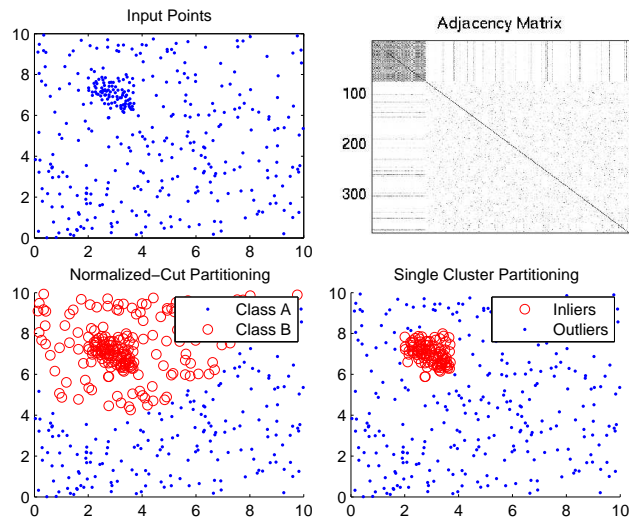

Fig. 1. K-way versus 1-way clustering. When attempting to identify a single cluster, k-way algorithms like Normalized Cuts perform poorly in comparison to the 1-way algorithm described in this paper.

algorithm. Clearly, attempting to produce "balanced" clusters produces poor results.

This paper will describe our algorithm, first considering the square/symmetric case, then generalizing the solution to non-symmetric problems. We then show how our single-cut graph partitioning algorithm can be used to solve several problems in robotics.

The main contributions of this paper are:

- Development and derivation of Single-Cluster Spectral Graph Partitioning (SCGP).
- Application of SCGP to outlier rejection, enabling a high fidelity Simultaneous Localization and Mapping (SLAM) result using range-only measurements, despite extreme noise.
- Application of SCGP to line fitting, demonstrating better performance than RANSAC.
- Application of SCGP to data association and robot localization, with promising results for future work.

II. RELATED WORK

Our approach is related to the existing body of work in spectral graph partitioning. Given an undirected, weighted

graph, existing algorithms identify two or more clusters such that nodes in each cluster are similar (large edge weights), and nodes in different clusters are dissimilar (small edge weights.) There are a variety of ways of expressing this idea numerically, leading to different partitioning algorithms with different metric functions. Ding's MinMaxCuts paper [2] contains a good survey of clustering methods, while the mathematical foundations of spectral clustering can be found in Fiedler and Donath's early papers [3]–[6].

Perona and Freeman explored machine-vision problem of extracting foreground features from non-salient background features [7]. Both Perona's approach and SCGP use the same eigenvector of the adjacency matrix in their solution, but use different means to find clusters. The algorithm presented here is also more general, in that it has been extended to handle the non-symmetric, non-square affinity matrices that arise in several problem domains.

Our approach to dealing with non-symmetric and non-square affinity matrices is similar to the approach described by Zha and He [8], in which they explore conventional partitioning algorithms in the context of document classification.

Single-cluster graph partitioning can be used in many domains in which Random Sample Consensus (RANSAC) [9] is often used. RANSAC forms candidate hypotheses in a randomized way, counting how many samples from the data set agree with each hypothesis within a preset threshold. The largest set of consenting samples is then used to fit a solution.

Another application of SCGP is data association, for which a number of popular algorithms exist. Neira's Joint Compatibility Branch and Bound (JCBB) [10] performs a tree search over all possible data associations, leading to an exponential run time. JCBB differs from other tree searches by adding a "compatibility" test at each node in order to reduce the search space.

Outlier rejection is often performed by gating measurements with a prior, or by using a median window. Gating with a prior simply means that measurements that are too unlikely given the current estimate of the process are discarded. Median windows consider a set of measurements in which the value is not expected to change considerably, discarding those measurements that are far from the median. We will present an outlier detection approach that does not require a prior and that can work even when a median window would fail.

III. Algorithm Overview

A. Formulation

We can view a set of N hypotheses and their pairwise-consistencies as a graph. Each hypothesis becomes a node, and the weight of an edge connecting two nodes is the pairwise consistency of those two hypotheses, $C(i, j)$. The $N \times N$ adjacency matrix is then:

$$A_{ij} = C(i, j) \tag{1}$$

Our goal is to find a set of hypotheses that, in some sense, are maximally consistent with each other. In terms of the graph, we want to find a set of nodes that are connected by edges with large weights.

We represent a set of nodes with an $N \times 1$ binary-valued *indicator vector*. For an indicator vector u, u_i is 1 if the ith node is a member, and zero otherwise. There is a large potential search space of indicator vectors; each node can either be in the set or not, yielding 2^N possible partitionings.

In order to find a set of "maximally consistent hypotheses", we must define a means of comparing the consistency of one set of hypotheses with another. The metric should have two basic properties:

- It should grow with the number of edges in the inlier cluster. These edges indicate increasing consistency among the inlier nodes.
- It should be penalized by the number of inliers, so that new nodes are only added when it is sufficiently desirable to do so. Otherwise, a trivial solution of setting the inlier set to be the entire candidate set can result.

In this paper, we use a simple metric that has a strong intuitive appeal. Namely, we attempt to maximize the *average consensus*– the sum of the edge weights that connect a candidate inlier node to the rest of the inlier cluster:

$$r(u) = \frac{u^T A u}{u^T u} \tag{2}$$

The quantity $u^T A u$ is the sum of all the edge weights in the subgraph containing the nodes in u, and the quantity $u^T u$ is the number of nodes in that subgraph.

This metric function rewards clusters with high self-consistency, while discouraging the addition of nodes which do not contribute to intra-cluster consistency. It also ignores nodes and edges belonging to outliers. This is important since we do not generally know how the outliers will be distributed; we do not want a randomly-occurring cluster of outliers influence the inlier set.

To further our intuition, consider a partitioning problem with boolean edge weights. Suppose we have a partial inlier set u, and want to determine whether node i should be added to the cluster. Suppose that before adding i, each inlier node is connected, on average, to three other nodes. Further suppose that node i is well-connected to the inliers, and that adding it to the set increases the average consensus to four. Clearly, adding node i makes the inlier set a better cluster. Conversely, if adding node i *decreases* the average consistency, then the inlier set is better without it. Unlike this example, our algorithm does not make these decisions sequentially. However, the intuition behind the algorithm remains.

B. Solution

Relaxation to continuous-valued indicator vector: There is no known polynomial-time solution to maximize $r(u)$ when u is a discrete-valued indicator vector. However, an approximate solution can be found by relaxing the constraint on u, allowing its elements to take any positive real value.

The extrema of $r(u)$ can be computed by setting the gradient of $r(u)$ to zero. Remembering that A is symmetric:

$$\nabla r(u) = \frac{Auu^T u - u^T Auu}{(u^T u)^2} = \frac{Au - r(u)u}{u^T u} = 0 \quad (3)$$

$$Au = r(u)u. \quad (4)$$

This is easily recognized as an eigenvector problem with an eigenvalue of $r(u)$. The maximum attainable value of $r(u)$ is the dominant eigenvalue of matrix A, which occurs when u is the dominant eigenvector. We note that Eqn. 2 is also known as the Rayleigh Quotient of matrix A. Provided that $A_{i,j} \geq 0$ (a natural property of virtually any pairwise consistency function), the Perron-Frobenius theorem guarantees that the maximum eigenvalue of A, and its eigenvector u, are both positive [11].

Discretization of the continuous indicator vector: The continuous-valued indicator vector can be interpreted as the importance of each hypothesis to the optimal set, where the set allows partial membership. For some applications, this ranking of importances might be sufficient. In general, however, it is desirable to convert the continuous indicator vector into a discrete one, resulting in a set v with boolean membership.

Our strategy is to pick a threshold t, computing v as:

$$v_i(t) = \begin{cases} 1 & \text{if } u_i \geq t \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

If the size of the desired set is known in advance, then the threshold can be computed such that v contains the desired number of hypotheses. This can be the case when there is prior information on the number of incorrect hypotheses.

In the general case, we do not know how many elements the set should have; we wish to determine this automatically. We propose two possible methods, both of which threshold u by searching for a scalar threshold t, but using different metric functions.

The simplest approach is to pick v such that the direction of v is as close as possible to u, i.e.:

$$v = \operatorname{argmax}_v \left(\frac{v^T u}{||v||} \right) \quad (6)$$

While the obvious method of maximizing Eqn. 6 is $O(N^2)$, v can be found in only $O(N \log N)$ time. Only N thresholds need to be tested, since u can have at most N distinct values. Given the elements of u in descending order, we can incrementally compute possible values of Eqn. 6 in $O(1)$ time each. Since there are N elements, we complete the search in $O(N)$ time. But we required the elements of u to be sorted, which requires $O(N \log N)$ time.

It can be shown that finding the threshold in this way is equivalent to maximizing $u^T B u / u^T u$, where B is the rank-1 approximation of A, $B = uu^T$.

The main appeal of this approach is its runtime cost, but it also preserves more information about the direction of the eigenvectors. Preliminary work shows that this is helpful when considering multiple-eigenvector variants of SCGP.

Another approach for picking a discretization threshold is to search for the v that maximizes $r(v)$. Since this is the metric that originally motivated SCGP, it makes sense to choose the v that maximizes it.

There are again N thresholds that must be considered, and the obvious searching algorithm performs an $O(N^2)$ multiply for each one, for a total complexity of $O(N^3)$. However, we can reduce this complexity to $O(N^2)$ by sorting the elements of u and incrementally computing the value of $r(v)$ as t decreases. Consider iteration n ($n \in [1..N]$) of this process; we can write $v_n = v_{n-1} + w_n$ where w_n is an indicator vector such that w_j is 1 iff $u_j = t_n$.

The denominator of Eqn. 2 is just the number of elements in v_n, which is trivially updated at each step. Let us just consider the numerator as a function of v, i.e., $N(v)$:

$$
\begin{aligned}
v_n &= v_{n-1} + w_n \\
N(v_n) &= v_n^T A v_n \quad (7) \\
N(v_n) &= (v_{n-1} + w_n)^T A (v_{n-1} + w_n) \\
&= v_{n-1}^T A v_{n-1} + v_{n-1}^T A w_n + w_n^T A v_{n-1} + w_n^T A w_n \\
&= N(v_{n-1}) + \sum_{i \in w, j \in v} 2A_{i,j} + \sum_{i \in w, j \in w} A_{i,j} \quad (8)
\end{aligned}
$$

Each time we increase the threshold, we must compute Eqn. 8, but the cost is only $O(N)$. Since we must perform N iterations, the total cost is $O(N^2)$. Alternatively, we can see that the cost is $O(N^2)$ since every element of A is included exactly once after all N iterations ($N(v_N) = \sum A_{i,j}$). The cost of sorting the elements of u is subsumed by the $O(N^2)$ cost.

While the continuous indicator vector u is provably the optimal continuous indicator vector, we cannot be certain that v will be the optimal discrete-valued indicator. The global optimum may not correspond to *any* thresholding of u. However, like other spectral approximations to NP-difficult problems, the performance of the algorithm is usually very good.

IV. ANALYSIS AND OPTIMIZATIONS

A. Rapid calculation of the first eigenvector

Fortunately, it is not necessary to perform a slow eigenvalue decomposition on matrix A in order to extract the maximum eigenvector; we can rapidly compute just the first eigenvector in $O(N^2)$ time. The behavior of $A^n x$ is dominated by the behavior of the largest eigenpair of A. The Power Method exploits this to compute the largest eigenvector by repeatedly left-multiplying a random vector by A. The product will converge to the eigenvector if $\lambda_1^n \gg \lambda_2^n$, which is typically the case. In most cases, we have found that performing only two or three iterations provides enough precision to find a good solution; we do not need many significant digits in order to perform the thresholding accurately. However, the Power Method can converge slowly if the eigenvalues are close in magnitude. The Inverse Power Method, coupled with shifting, can accelerate convergence in this case [12].

The relative magnitude of the secondary eigenvalues conveys information about the ambiguity in the problem. If two different eigenvectors have similar magnitudes, it means that there are two equally-good explanations of the data (e.g., two sets of inlier clusters). SCGP, as presented in this paper, is intended to deal with only one set of inliers. For many applications (where there can only be one true inlier cluster), the relative magnitude of the second eigenvalue can serve as a warning that the reliability of the results may be low.

If the certainty of a clustering is low, it still might be of use to systems which can cope with uncertainty. For example, particle filters such as Montemerlo's FastSLAM [13] can simultaneously track multiple incompatible hypotheses; particles incorporating incorrect hypotheses tend to have very low likelihoods and thus die out. Leonard's delayed-state filter [14] could simply delay making a decision until further evidence was available.

B. Pairwise consistency test considerations

At the heart of the SCGP is the pairwise consistency test which is used to construct the adjacency matrix. The behavior of the pairwise test can have a profound effect on the success or failure of the algorithm.

The robustness of SCGP is particularly sensitive to any "DC offset" in the adjacency matrix. In other words, it is beneficial for inconsistent pairs to have a consistency that is not only smaller than consistent pairs, but as close to zero as possible. We illustrate this importance with a simplified scenario.

Suppose there are N_t true hypotheses and N_f false hypotheses. Assume that the pairwise consistency test reliably assigns edge weights of C_t to pairs of true hypotheses, C_f to all other pairs.

In this special case, if we assume that $C_t \geq C_f$, the optimal solution is either the cluster of true hypotheses, or the cluster consisting of all points. We wish to determine what constraint(s) on C_t and C_f will ensure that the cluster of true hypotheses will have a greater eigenvalue. We begin by writing down the inlier metrics for the two cases:

$$r(u_t) = \frac{C_t N_t^2}{N_t} \tag{9}$$

$$r(u_t + u_f) = \frac{C_t N_t^2 + C_f N_f^2 + 2C_f N_t N_f}{N_t + N_f} \tag{10}$$

The SCGP will produce the correct value so long as $r(u_t) > r(u_t + u_f)$. This will be true so long as:

$$\frac{C_t}{C_f} > \frac{N_f}{N_t} + 2 \tag{11}$$

In other words, the ratio C_t/C_f determines how many outliers can be tolerated. Thus, a good consistency metric should try to make C_f as close to zero as possible, in order to maximize C_t/C_f.

A different consideration for the pairwise consistency test is the weight of a hypothesis with itself. If we change the self-consistency of hypotheses, it is equivalent to adding a multiple of the identity matrix to the adjacency matrix.

$$(A + \alpha I)u = r(u)u$$
$$Au = (r(u) + \alpha)u \tag{12}$$

We see immediately that the eigenvectors (and thus the SCGP solution) are unaffected. However, α shifts the eigen*values*, which can help or hinder the convergence of Power Method-style algorithms for estimating the eigenvectors.

C. Generalization to rectangular matrices

SCGP can be generalized to non-square and non-symmetric matrices, a situation that arises in many problems. For example, given a set of M line hypotheses and N points, we can construct an $M \times N$ affinity matrix A where A_{ij} is some measure of the consistency between line i and point j. Our goal is to extract *two* indicator vectors: an $M \times 1$ vector for the lines, and an $N \times 1$ vector for the points.

While A cannot be directly interpreted as the adjacency matrix of a graph, we can still construct a graph that captures the relationship between lines and points. Lines and points have edges between them with weights according to the elements of A. But there are no edges between lines, and no edges between points. We can write the adjacency matrix for the resulting bipartite graph as:

$$P = \left[\begin{array}{c|c} 0 & A \\ \hline A^T & 0 \end{array} \right] \tag{13}$$

SCGP can be performed exactly as before on this modified matrix. The standard SCGP solution will yield a single $(M + N) \times 1$ indicator vector w, which can easily shown to be the concatenation of the desired $M \times 1$ vector u and the $N \times 1$ indicator vector v. SCGP on matrix P yields:

$$\frac{w^T P w}{w^T w} = \frac{[u^T v^T] \left[\begin{array}{c|c} 0 & A \\ \hline A^T & 0 \end{array} \right] \left[\begin{array}{c} u \\ v \end{array} \right]}{[u^T v^T] \left[\begin{array}{c} u \\ v \end{array} \right]} = \frac{2u^T A v}{u^T u + v^T v} \tag{14}$$

If we write A according to its Singular Value Decomposition (SVD), this expression becomes:

$$\frac{2u^T A v}{u^T u + v^T v} = \frac{2u^T U S V^T v^T}{u^T u + v^T v} \tag{15}$$

From this we can see that the expression is maximized by setting u and v to be the dominant left and right eigenvectors of A (not P). This is notable; we can solve general SCGP problems without constructing matrix P.

The dominant eigenpair of A can be computed by using a variation of the Power Method. The following procedure can be easily derived: beginning with a random u_0, compute $v_{n+1} = A^T u_n$ and, in turn, $u_{n+1} = A v_n$. The convergence rate is again determined by the ratio of the magnitudes of the two largest eigenvalues.

The resulting vectors could correspond to either a positive or negative eigenvalue, depending on the initialization of the Power Method. Assuming that $A_{i,j} \geq 0$, the Perron-Frobenius theorem [11] again guarantees that the dominant eigensystem can be written with a positive eigenvalue and positive left and right eigenvectors; it is these positive vectors that we want. If the power method produces a negative vector, it means that the corresponding eigenvalue was also negative: the minus signs cancel in the eigensystem. This means that if we encounter a negative vector, we should multiply it by -1 before computing the discrete indicator vector. Alternatively, we can simply initialize the power method with random positive vectors; since A is nonnegative, each iteration will preserve the positivity of the vectors.

V. APPLICATIONS AND RESULTS

We present three different applications of SCGP. These applications demonstrate not only how a variety of robotics problems can be mapped onto the SCGP algorithm, but also the quality of the results.

A. Outlier Rejection for Range-Only SLAM

We demonstrated in [15] a SLAM system operating with range-only data on an Autonomous Underwater Vehicle (AUV). With only range (and *not* bearing) information to landmark features, SLAM becomes quite difficult. Newman explored the use of large-scale numerical optimizations over the entire robot trajectory and feature state [16]; unfortunately, their approach suffers from numerical instability. Kantor and Singh also discuss range-only SLAM in [17], however, their algorithm requires reasonable priors on feature locations.
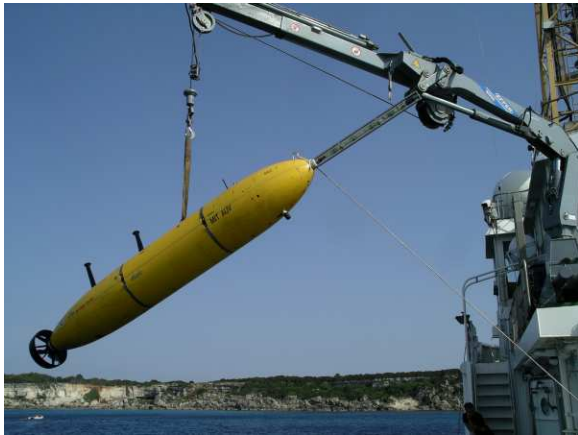

Fig. 2.   Odyssey Autonomous Underwater Vehicle. Using SCGP to reject outliers from the vehicle's range measurements to navigation beacons, we were able to implement a reliable SLAM system.

As pointed out in both [15] and [16], outlier rejection is critical in range-only SLAM. Given reliable data, the problem becomes relatively tractable. But in the underwater setting, explored by both of these papers, the noise in range measurements can be profound. In this section, we will elaborate on how SCGP was used, as well as present new navigation results

on a much longer and more difficult dataset than we did in [15].

Range measurements to stationary beacons were collected as the AUV maneuvered, but the range measurements were corrupted by noise due to interference from the AUV's Synthetic Aperture Sonar (SAS). While range measurements were collected every few seconds for two beacons, measurements for the other two beacons occurred much more rarely due to interference.
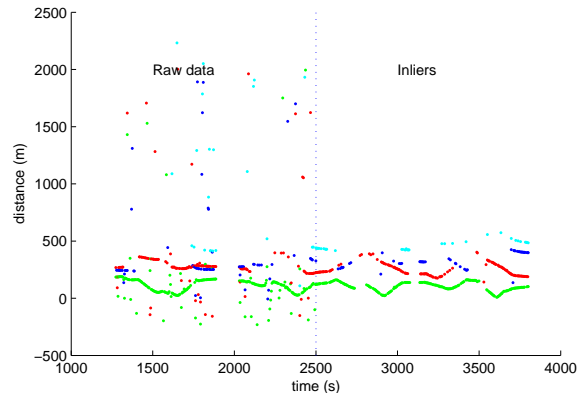

Fig. 3.   Outlier Rejection Results. The ranges to four stationary beacons are plotted during the course of a mission. Due to extensive interference from the vehicle's synthetic aperture sonar, range measurements were unreliable. Use of SCGP removed virtually all spurious data, without rejecting an excessive number of inliers. Notably, SCGP worked well even when data was very sparse. No prior on the positions or ranges to the beacons was used.

Outlier rejection is typically performed by rejecting points which differ too much from the prior. However, the goal of this system was to localize the beacons, thus no prior was available. SCGP provided a means of identifying a set of consistent samples, without the need for a prior.

For a set of N measurements, we form N hypotheses: hypothesis i asserts that measurement i was an inlier. Restricting ourselves to the 2D case, a single range measurement constrains the location of the beacon to a circle around the vehicle's current position. Two range measurements form two circles, and these circles might or might not intersect. Two measurements are consistent ($A_{ij} = 1$) if the circles intersect, indicating a possible solution for the beacon's location (see Fig. 5). If the circles do not intersect, we set $A_{ij} = 0$. The vehicle's true position is not actually available, so we use the vehicle's dead-reckoned position when computing the consistency function.

In other words, the consistency function measures whether it is *possible* for the two measurements to be simultaneously true, without needing to actually determine what solution (or solutions) that would imply. This boolean consistency function is quite weak; it is quite easy for outliers to intersect with other measurements. However, the inliers are better connected than the outliers, and so SCGP is able to produce good results.

The filter can be operated in a causal manner, classifying measurements as inliers/outliers in real time, by using a sliding window including the most recent measurements.
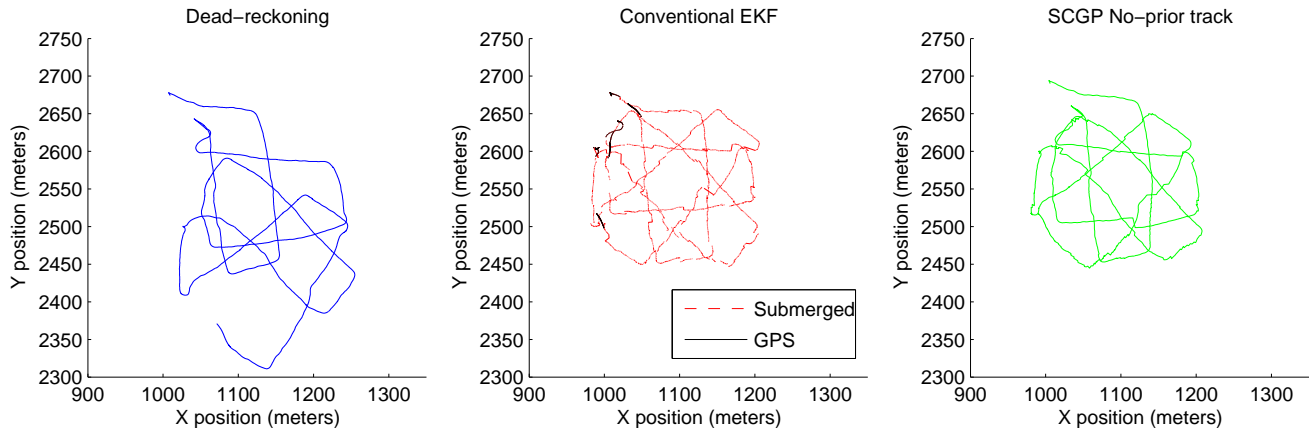
Fig. 4. SCGP-based Range-Only SLAM. The AUV's dead-reckoned position is quite poor (left). In middle, a conventional Extended Kalman Filter (EKF), with prior knowledge of beacon locations, serves as our baseline 'ground truth'. The SCGP-based SLAM algorithm, using neither GPS nor prior information on beacon locations, closely resembles the conventional EKF.
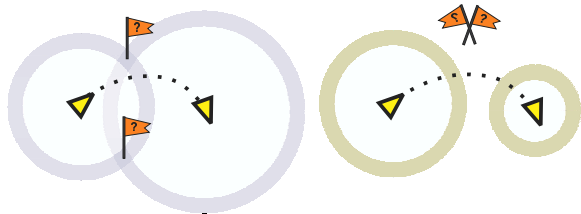

Fig. 5. Consistent and Inconsistent Range-Only Measurements. Left: Two range measurements are consistent if they overlap (i.e., have a simultaneous solution). If the measurements do not overlap (right), they are inconsistent.

The pairwise consistency test was able to incorporate information about the vehicle's motion to help reject outliers. Not only did measurements have to be consistent with each other (which, for an intersection test, requires little more than that range measurements vary slowly), the change in range had to be consistent with the motion of the vehicle. Consequently, the filter could reject "bunches" of outliers which a median filter might accept. Multi-path measurements, in which the measured range varies at some multiple of the vehicle's speed, are also reduced.

Once the inlier range measurements were isolated, implementing a SLAM filter was relatively straightforward. A Hough Transform-style algorithm was used to initialize new features, after which a simple EKF incorporated additional operations. The full details of this algorithm are described in our earlier paper [15]. Note that only four range transponders were deployed for this experiment.

B. Line Fitting

Using the generalized rectangular case, SCGP can be used to robustly fit points to a line. In this problem, we assume we have a large number of points, some fraction of which lie near a single line. Our goal is to discover the line.

A pairwise consistency function between points, in this case, is not revealing: any two points form a perfectly valid line. Instead, our solution is to generate a number of line hypotheses by randomly selecting points. We can then compute the consistency of each line hypothesis with each point, forming a rectangular affinity matrix. If d is the distance from a point to a line, the pairwise consistency of the point and line is:

$$A_{ij} = e^{-d^2/2\sigma^2} \tag{16}$$

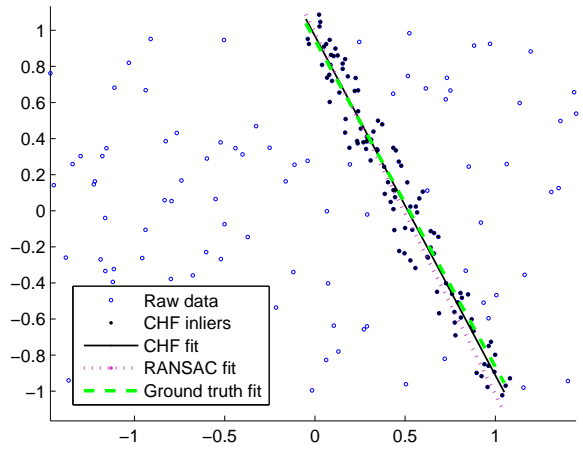We set σ according to an estimate of the noise magnitude, which gives our consistency function a natural interpretation as the probability that the point was drawn from a distribution of points around the line. We can safely omit scale factors, as they do not impact the eigenvectors.

If M is the number of line hypotheses we consider, and N is the number of points, our affinity matrix will be $M \times N$. We note that, for a given ratio of inliers to outliers, that M will need to grow as $O(N^2)$. The total runtime for SCGP will be $O(MN)$.

RANSAC is similar in several ways; it randomly generates line hypotheses and tests points against them. RANSAC simply returns the line which was consistent with the greatest number of points. Like SCGP, RANSAC has a complexity of $O(MN)$.

SCGP returns a set of points which are classified as inliers. We then perform a least-squares line fit to these points, producing a robust and accurate estimate. With SCGP, the inlier points are affected by every line hypothesis, whereas with RANSAC, only the best line hypothesis affects the line estimate. In RANSAC, all of the computation performed while evaluating other line candidates is discarded. This explains why SCGP can produce better results than RANSAC with the same asymptotic runtime complexity.

Fig. 6 shows a set of 200 2D points, 100 of which are noisy samples of an actual line. The remainder are uniformly distributed random points. Given the task of extracting the line, we have found that SCGP performs better than RANSAC for a given number of line hypotheses (and thus identical asymptotic complexity.) This is attributable to the SCGP's greater ability to estimate the set of points belonging to the line.

Fig. 6. Line Fitting Results. SCGP can be used to estimate lines in a method similar to, but more stable than, RANSAC. Given a fixed number of line hypotheses, SCGP consistently outperforms RANSAC with lower average error and less variance. A typical run is plotted. Each run contained 200 points, half of which were inliers corrupted by $\sigma = 0.1$ noise; the remainder were uniformly distributed noise. Both SCGP and RANSAC were limited to 40 hypotheses.

Our line fitting experiments show that not only is the asymptotic complexity of SCGP the same as RANSAC, but that they have similar coefficients. Over a broad set of problem sizes, our experiments showed that SCGP's runtime was between 0.996 and 1.251 times that of RANSAC.

C. Data Association

Data association, determining which observations correspond to the same physical features, is an essential component of most navigation systems and a key problem in Simultaneous Localization and Mapping (SLAM). Recognizing landmarks is a fundamental way of correcting accumulated navigation error.

Algorithms like Joint Compatibility, which perform exponentially complex searches over all possible data associations, can use an arbitrarily complex metric for evaluating a hypothesis set. But while full searches require exponential time (in the worst case), the SCGP can be computed in polynomial time. Further, the performance of the SCGP is quite good on data association problems, either providing a solution or serving as a smoke test to reduce the number of hypotheses which would be fed into a traditional search.

As a concrete example, consider a data association problem on corner features. Corners are a convenient landmark for navigation systems, since a data correspondence on corners uniquely and completely specifies a rigid-body transformation.

A typical data association problem is shown in Fig. 7. A robot is attempting to determine its position in a previously known map (with corners designated with letters), by matching corners currently visible to it (designated by numbers). We can
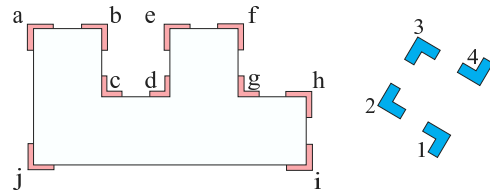

Fig. 7. Example Data Association/Robot Localization Problem. A global map (left) is composed of a number of corner features. A subset of those features are reobserved (right). SCGP can be used to solve the correspondence problem and localize the robot.

see that the correct data association solution is d1/e2/f3/g4. Note that any one of these data associations is sufficient to specify the rigid body transformation that aligns the robot's scan with the map. These four data associations all result in identical rigid-body transforms.

In order to map this problem onto SCGP, we create a hypothesis for every possible data association. Note that the figure contains both 90 degree and 270 degree corners, and we do not form hypotheses for corners of different types.

Twenty hypotheses are possible; corners 1 and 4 can each be matched to c, d, and g. Corners 2 and 3 can each be matched to a, b, e, f, h, i, and j. For this example, we define a boolean pairwise consistency function which is true if the rigid-body transformations implied by the two hypotheses are identical. In the 20×20 adjacency matrix, twelve hypotheses pairs are consistent; all other cells are zero. Six result from the true correspondences: d1e2, d1f3, d1g4, e2f3, e2g4, and f3g4. Three result from the laser scan partially matching the left most vertical area (abc): a2b3, a2c4, and b3c4. Similarly, three result from the partial match on the far right: g1h2, g1i3, and h2i3. In this simplified problem, we set $A_{ij} = 1$ if the data associations lead to the same rigid-body transformations, otherwise 0. In the more general case where measurements are corrupted by noise, a "softer" consistency function (most likely a probability) would instead be used.

The SCGP solution to this data association problem is shown in Fig. 8. The four correct hypotheses are strongly amplified by the SCGP.

It is also interesting to consider the case when the data association is, in fact, ambiguous. To see this, the same experiment–with corner 4 deleted– was run again. The eigenvalues in this case are also shown in Fig. 8. Without corner 4, there are two possible locations for the robot. The eigenvectors give these two solutions, and their eigenvalues are equal, indicating that the solutions are equally well supported.

The runtime of the SCGP on data association, with M features being mapped onto N features, is $O(M^2 N^2)$, since there are MN feature-correspondence hypotheses to consider. The resulting affinity matrix is $MN \times MN$.

It is possible to introduce null hypotheses into SCGP, though each null hypothesis adds a column and row to the matrix. To prevent a cluster of null hypotheses from forming a highly self-consistent set, null hypotheses should be considered to be inconsistent with other null hypotheses.
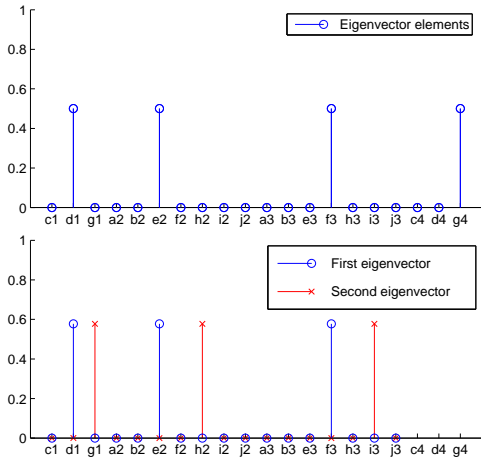
Fig. 8. Eigenvector elements for the data association problem in Fig. 7. The top figure corresponds to the problem shown in Fig. 7; the correct hypotheses d1, e2, f3, and g4 are given strong scores, with all other hypotheses scoring zero. In the second diagram, corner 4 has been removed, creating an ambiguity: the robot could either be in the middle or on the right. Not only is the situation ambiguous, but both situations are equally-well supported. The largest eigenvalue is repeated, and the corresponding eigenvectors represent both solutions.

VI. CONCLUSION

Single Cluster Graph Partitioning provides a useful method for extracting a set of consistent hypotheses from noise. In these problems, conventional k-way clustering algorithms generally perform poorly.

SCGP is the result of maximizing a simple heuristic: the average consistency of the inlier set. We showed how this maximization is performed, both in the square and symmetric case, and also extended this result to the general rectangular case. We also showed how to minimize computational complexity, and how to design a pairwise consistency metric for maximum performance.

When applied to several contemporary robotics problems, SCGP performs very well.

- In the context of range-only SLAM, SCGP can correctly identify inlier range measurements, enabling good mapping performance with a simple EKF.
- SCGP compares favorably to RANSAC, producing lower error and lower variance estimates with the same asymptotic computational complexity.
- SCGP can estimate solutions to data association problems in polynomial time.

This paper presented real-world outlier rejection results using SCGP. While the synthetic results of SCGP on line fitting and data association are promising, one area of future work is to test SCGP on additional real-world datasets. Another future goal is to examine the case when multiple inlier sets are present.

REFERENCES

[1] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, pp. 888–905, August 2000.

[2] C. Ding, X. He, H. Zha, M. Gu, and H. D. Simon, "A minmaxcut spectral method for data clustering and graph partitioning," Lawrence Berkeley National Laboratory, Tech. Rep. 54111, 2003.

[3] W. Donath and A. Hoffman, "Lower bounds for the partitioning of graphs," *IBM J. of Research and Developement*, vol. 17, pp. 420–425, 1973.

[4] M. Fiedler, "Algebraic connectivity of graphs," *Czecheslovak Mathematical Journal*, vol. 23, no. 98, pp. 298–305, 1973.

[5] ——, "A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory," *Czecheslovak Mathematical Journal*, vol. 25, no. 100, pp. 619–633, 1975.

[6] ——, "Laplacians of graphs and algebraic connectivity," in *Combinatorics and Graph Theory*, ser. Banach Center Publications, vol. 25, Warsaw, 1989, pp. 57–70.

[7] P. Perona and W. Freeman, "A factorization approach to grouping," *Lecture Notes in Computer Science*, vol. 1406, pp. 655–670, 1998. [Online]. Available: citeseer.ist.psu.edu/perona98factorization.html

[8] H. Zha, X. He, C. Ding, H. Simon, and M. Gu, "Bipartite graph partitioning and data clustering," in *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*. ACM Press, 2001, pp. 25–32.

[9] M. A. Fischer and R. C. Bolles, "A paradigm for model-fitting with applications to image analysis and automated cartography," 1981.

[10] J. Neira and J. D. Tardos, "Data association in stochastic mapping using the joint compatibility test," *IEEE Transactions on Robotics and Automation*, December 2001.

[11] R. A. Horn, *Topics in matrix analysis*. New York, NY, USA: Cambridge University Press, 1986.

[12] G. Strang, *Introduction to Linear Algebra*. Wellesley-Cambridge Press, 1993.

[13] M. Montemerlo, "Fastslam: A factored solution to the simultaneous localization and mapping problem with unknown data association," Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, July 2003.

[14] J. J. Leonard and R. J. Rikoski, "Incorporation of delayed decision making into stochastic mapping," in *Proceedings of Autonomous Underwater Vehicles*, 2000, pp. 533–542.

[15] E. Olson, J. Leonard, and S. Teller, "Robust range-only beacon localization," in *IEEE Autonomous Underwater Vehicles (AUV '04)*, 2004.

[16] P. M. Newman and J. Leonard, "Pure range-only sub-sea SLAM," in *Proceedings of the IEEE Conference on Robotics and Automation (ICRA '02)*, 2003.

[17] G. A. Kantor and S. Singh, "Preliminary results in range-only localization and mapping," in *Proceedings of the IEEE Conference on Robotics and Automation (ICRA '02)*, May 2002.