暹罗·J·擎天柱。

第25卷，第1期，第468-501页

2015工业与应用数学学会

**基于半定规划的多点云全局配准**∗

K、 N.CHAUDHURY†，Y.KHOO‡，和A.SINGER§

**抽象的。**考虑Rand M局部坐标系中通过未知刚性变换相关的N个点。对于每一个点，我们在一些坐标系中给出了它的局部坐标的测量值（可能有噪声）。或者，对于每个坐标系，我们观察点子集的坐标。在分子构象和传感器网络定位的分布式方法中，以及计算机视觉和图形学中，都出现了从这些测量值估计N个点的全局坐标（直到一个刚性变换）的问题。这个问题的最小二乘公式虽然是非凸的，但当M=2时（基于奇异值分解（SVD））有一个众所周知的闭式解。然而，对于M≥3，没有已知的闭式解。在本文中，我们证明了如何将最小二乘公式化为凸规划，即半定规划（SDP）。通过建立SDP的唯一性与刚度理论结果之间的联系，我们证明了SDP松弛精确稳定恢复的条件。特别地，我们证明了与先前提出的光谱松弛相比，SDP弛豫可以在更不利的条件下保证恢复，并且推导了SDP松弛引起的配准误差的误差界。我们还提供了模拟数据的数值实验结果，以证实理论结果。我们从经验上证明：（a）与谱松弛不同，对于SDP（即，我们能够解决原始的非凸最小二乘问题）的松弛间隙在某个噪声阈值以下大多为零，（b）SDP的性能明显优于谱优化和流形优化方法，尤其是在大噪音水平下。*d*

**关键词。**全局配准，刚性变换，刚性理论，谱松弛，谱间隙，凸松弛，半定规划（SDP），精确恢复，噪声稳定性

**AMS学科分类。**90C22、52C25、05C50

**多伊。**10.1137/130935458

**1介绍。**点云注册问题出现在计算机视觉和图形学[52，59，65]和分布式分子构象方法[20，17]和传感器网络定位[16，9]中。所讨论的配准问题是根据（可能有噪声）较小的点云子集（称为斑块）P1，…，PM的坐标知识来确定点云P的坐标，这些点云子集是通过一些一般变换从P导出的。在某些应用[45，59，40]中，人们通常对寻找一致对齐P1，…，PM的最佳变换（每个面片一个）感兴趣。这可以看作是确定P[16，51]坐标时的一个子问题。



\*编辑于2013年9月3日收到；2014年12月22日接受出版（修订版）；2015年3月4日电子版出版。

http://www.siam.org/journals/siopt/25-1/93545.html

†印度科学院电气工程系，印度班加罗尔560012(kunal@ee.iisc.ernet.in). 本文作者的研究得到了瑞士国家科学基金会的部分资助，资助项目为PBELP2-135867，并获得了普林斯顿大学国立普通医学科学研究所的R01GM090200奖，最近还获得了印度科学研究所的创业资助。

‡普林斯顿大学物理系，新泽西州普林斯顿08540(ykhoo@princeton.edu). 作者的研究得到了美国国立卫生研究院R01GM090200-01的部分支持。

§普林斯顿大学数学与PACM系，新泽西州普林斯顿08540（amits@数学普林斯顿教育大学). 本文作者的研究得到了AFOSR的FA9550-13-1-0076和FA9550-12-1-0317、美国国家普通医学科学研究所的R01GM090200奖和西蒙斯基金会的LTR DTD 06-05-2012奖的部分支持。

468

在本文中，我们考虑刚性配准问题，其中给定Pi内的点通过未知的刚性变换从P（理想）获得。此外，我们假设局部补丁和原始点云之间的对应关系是已知的，也就是说，我们预先知道给定的Pi中包含来自P的哪些点。事实上，在分子构象的分布式方法和传感器网络定位[9，69，16]中，我们可以控制对应关系。虽然这种对应关系不能直接用于某些图形和视觉问题，例如多视图配准[49]，但原则上可以通过对齐一对面片来估计对应关系，例如使用ICP（迭代最近点）算法[6，51，36]。

**1.1条。两个补丁注册。**两个补丁注册的特殊问题已经得到了很好的研究[21，34，2]。在无噪声背景下，我们给出了R中的两个点云{x1，…，xN}和{y1，…，yN}，其中后者是通过对前者的一些刚性变换得到的。也就是说，*d*

（1.1）yk=Oxk+t（k=1，…，N），

其中O是某个未知的d×d正交矩阵（满足OT O=Id），t∈是一些未知的平移。*d*

问题是从上面的方程中推断出O和t。为了唯一地确定O和t，必须至少有N≥d+1个非退化点。[1]在这种情况下，可以简单地通过固定（1.1）中的第一个方程并从中减去（以消除t）任何剩余的d方程来确定O。假设我们减去下一个d方程：

[y2−y1···y+1−y1]=O[x2−x1···x+1−x1]；*dd*

在非退化假设下，O右边的矩阵是可逆的，这就得到了O。

在实际设置中，（1.1）仅适用于，例如，由于噪音或模型缺陷。一种特殊的方法是通过考虑以下最小二乘法程序来确定最佳O和t：

（1.2）最小值。

请注意，由于优化域是O（d）×R，这是非凸的，所以这个问题看起来很困难。值得注意的是，这个非凸问题的全局极小值可以精确地找到，并且有一个简单的闭式表达式[19,39,32,21,34,2]。更精确地说，最优O由vut给出，其中U∑vt是*d*

其中是各自点云的质心。最好的翻译是。

两个补丁注册有一个封闭形式的解决方案这一事实被用于所谓的增量（顺序）方法来注册多个补丁[6]。最著名的方法是ICP算法[51]（请注意，ICP除了注册相应的点外，还使用了其他启发式和改进方法）。大体上，序列配准的思想是一次注册两个重叠的块，然后用某种方法对估计的成对变换进行积分。集成可以在本地（逐片）或使用基于全局周期的方法（如同步）来实现[52、35、53、59、63]。最近，有研究表明，通过局部注册重叠的补丁，然后利用同步技术集成成对变换，可以设计出高效且健壮的分布式传感器网络定位方法[16]和分子构象[17]。注意，当注册阶段是局部的时，同步方法以全局一致的方式集成局部变换。这使得它对经常困扰局部积分方法的误差传播具有鲁棒性[35，63]。

**1.2条。多匹配注册。**为了描述多匹配注册问题，我们首先介绍了一些符号。假设x1，x2，…，xN是R中一个点云的未知全局坐标。该点云被划分为P1，P2，…，PM，其中每个Pi{x1，x2，…，xN}。这些面片通常是重叠的，因此一个给定的点可以属于多个面片。我们用无向二部图Γ=（Vx∪VP，E）来表示这种成员关系。顶点集合Vx={x1，…，xN}表示点云，而VP={P1，…，PM}表示面片。边集E=E（Γ）连接Vx和VP，满足（k，i）∈E当且仅当xk∈Pi时给出。此后，我们将把Γ称为成员关系图。*d*是

在本文中，我们假设一个给定面片的局部坐标可以（理想地）通过一个刚性变换（即通过一些旋转、反射和平移）与全局坐标相关联。更准确地说，对于每个片Pi，我们将一些（未知）正交变换Oi和平移ti相关联。如果点xk属于patch Pi，则其在Pi中的表示由（cf.（1.1）和图1给出

（1.3）xk，i=OiT（xk−ti）（k，i）∈E（Γ）。

或者，如果我们修复了一个特定的补丁Pi，那么对于属于该补丁的每个点，

（1.4）xk=Oixk，i+ti（k，i）∈E（Γ）。

特别是，给定点可以属于多个面片，并且在每个面片的坐标系中有不同的表示。

本文的前提是给出隶属度图和局部坐标（称为测量），即，

（1.5）Γ和{xk，i，（k，i）∈E（Γ）}，

目标是从（1.5）中恢复坐标x1，…，xN以及在此过程中未知的刚性变换（O1，t1），…，（OM，tM）。请注意，全局坐标是根据全局旋转、反射和平移来确定的。我们说两个点云（也被称为配置）是一致的，如果一个是通过另一个的刚性变换得到的。我们总是将两个完全相同的构型确定为一个构型。

在对测量值进行适当的非退化假设的情况下，一项任务将是在Γ上指定适当的条件，在此条件下可以唯一地确定全局坐标。直觉上，很明显补丁必须有足够的

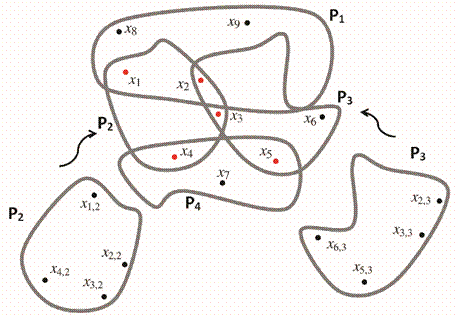


图1。在R（1.5）上注册三个补丁以表示局部坐标的问题）。有助于注册的是属于两个或多个补丁（用红色标记）的公共点。注意，在这种情况下，顺序或成对注册将失败。这是因为没有一对面片可以注册，因为它们的公共点少于三个（至少需要三个点来修复R中的旋转、反射和平移）2*，其中需要从相应的局部面片坐标中找到点的全局坐标。面片P中点的局部坐标*2 *还有P*3 *如图所示（参见*2*). 本文提出的基于SDP的算法*

*执行全局注册，并能够恢复此示例的精确全局坐标。*

共同点对于注册问题有一个独特的解决方案。例如，很明显，如果Γ断开连接，则无法唯一地恢复全局坐标。

在实际应用中，我们会遇到噪声环境，其中（1.4）只适用于大约。在这种情况下，我们想确定全局坐标和刚性变换，使（1.4）中的差异最小。我们特别考虑以下二次损失：

（1.6），

其中·是R上的欧几里德范数。优化问题是针对以下变量最小化φ：*d*

*十*1*，x*2*，…，xN*∈R*d，O*1*，…，OM*∈O（d） ，t1，…，tM∈R*d.*

问题的输入包括（1.5）中的测量值。注意，我们的最终目标是确定x1，x2，…，xN；刚性变换可以看作是潜在变量。

多匹配配准问题本质上是非凸的，因为需要在正交变换的非凸域上进行优化。针对这一问题，人们采用了与优化文献不同的思想，包括拉格朗日优化法和投影法。在拉格朗日设置中，正交性约束被纳入目标；在投影法中，约束在优化的每一步之后都被强制执行[49]。在观察到注册问题可以看作是Grassmanian和Stiefel流形上的一个优化问题之后，研究人员利用流形优化理论和实践中的思想提出了算法[40]。对这些方法的详细回顾超出了本文的范围，相反，我们建议感兴趣的读者参考这些优秀的评论[18，1]。然而，基于流形的方法本质上是局部的，不能保证找到全局最小值。此外，这类方法的噪声稳定性很难得到验证。

**1.3条。贡献。**本文的主要贡献可归纳为以下几类。

1.    *算法*. 我们演示了如何将转换从（1.6）中分解出来，从而将最小二乘问题简化为以下优化：

                 （1.7）最大）根据O1，…，OM∈O（d），

其中Cij∈R*d*×天（1≤i，j≤M）是某些Md×Md的半正定块矩阵C的第（i，j）个子块。给出（1.7）的解，就可以简单地通过求解线性方程组得到所需的全局坐标。对于大规模问题（1），几乎不可能找到（1.7）的全局最优值，因为这涉及到在一个巨大的非凸参数空间上对二次成本进行优化。事实上，最简单的情况是d=1，其中C是拉普拉斯矩阵，对应于最大割问题，这是众所周知的NP难问题。本文的主要结论是（1.7）可以被放宽为一个凸规划，即半定规划（SDP），其全局最优值可以用标准现成的求解器在多项式时间内逼近到任意有限精度。这为算法2中描述的全局注册提供了一个可处理的方法。由[40]中已经考虑过的（1.7）的谱松弛导出的相应算法在算法1中作了描述，以供参考。

2.    *精确回收*. 利用算法2给出了（1.7）中系数矩阵C的精确恢复条件。特别地，我们证明了算法2的精确恢复问题可以映射为刚性理论问题，这些问题在[68，25]中已经被研究过了。本节的贡献在于（1.7）中的C矩阵与本文中考虑的各种刚性概念之间的联系。我们还提出了一个有效的C的随机秩检验，可以用来证明确切的恢复（由[31，26，54]中的工作激励）。

3.    *稳定性分析*. 我们研究了算法1和算法2的稳定性，在噪声模型中，使用有界大小的噪声来扰动斑块坐标（注意，在[40]中没有研究谱松弛的稳定性）。我们的主要结果是定理5.2，它指出如果C满足一个特定的秩条件，那么算法2的注册误差在噪声级的一个常数因子内。据我们所知，目前还没有一个多匹配注册算法具有类似的稳定性保证。

4.    *实证结果*. 我们在模拟数据上给出数值结果，以数值验证算法1和算法2的精确恢复和噪声稳定性。我们的主要实证结果如下：

（1） 在重建质量方面，半定松弛比基于谱和流形的优化（例如，以谱解作为初始化）表现得更好（参见图7中的第一个图）。

（2） 在一定的噪声水平下，我们实际上能够使用半定松弛法来解决原始的非凸问题（参见图7中的第二个图）。

**1.4条。更广泛的背景和相关工作。**目标（1.6）是两个补丁的目标的直接扩展[19，21，34，2]。事实上，这一目标在张等人之前就已经考虑过了。分布式传感器定位[69]。目前的工作也与Cucuringu等人的工作密切相关。关于分布式本地化[16,17]，其中类似的目标被隐式优化。这些工作的共同主题是，一旦通过某种方式确定了局部坐标，就使用某种形式的优化方法来全局注册这些面片。然而，在实际执行优化的各种算法之间有一些根本的区别。Zhang等人。[69]使用交替最小二乘法对全局坐标和变换进行迭代优化，据我们所知，这没有收敛保证。另一方面，Cucuringu等人。[16，17]首先对正交变换进行优化（使用同步[53]），然后使用最小二乘法拟合求解平移（实际上是全局坐标）。在这项工作中，我们将这些不同的想法结合到一个单一的框架中。虽然我们的目标与[69]中使用的目标相似，但我们共同优化刚性变换和位置。特别是，第2节中考虑的算法避免了[69]中与交替最小二乘法相关的收敛问题，并且能够注册不能使用[16，17]中的方法注册的补丁系统。

另一个密切相关的工作是Krishnan等人的论文。在全局注册[40]中，通过将（1.2）中的目标扩展到多匹配情况来计算最佳变换（具体地说是旋转）。随后的数学公式与我们的公式非常相似，事实上，会导致类似于（1.7）的子问题。Krishnan等人。[40]提出利用流形优化来求解（1.7），其中流形是旋转的乘积流形。然而，如前所述，流形方法通常不能保证收敛性（达到全局最小值）和稳定性。此外，（1.7）中的歧管未连接。因此，如果初始猜测在流形的错误分量上，任何局部方法都无法获得（1.7）的全局最优解。

正是在这一点上，我们离开了[40]，也就是说，我们建议放松

（1.7）变成可操纵的SDP。这是由于在使用

非凸（特别是NP难）问题的SDP松弛。例如，见[43，23，66，47，12，41]和评论[60，48，70]。注意，对于d=1，（1.7）是一个二次布尔优化，类似于MAX-CUT问题。Goemans和Williamson[23]的开创性工作提出了一种基于SDP的随机取整算法来求解MAX-CUT。我们在第2节中考虑的半定松弛就是由这项工作所激发的。结合目前的工作，我们注意到可证明稳定的SDP算法已经被考虑用于低秩矩阵完成[12]、相位检索[13,62]和图局部化[37]。

我们注意到这里讨论的注册问题的一个特例就是所谓的广义Procrustes问题[28]。在刚刚介绍的点补丁框架内，Procrustes分析的目标是找到最小化的O1，…，OM∈O（d）

（1.8）*.*

换言之，目标是通过正交变换实现M片的最佳对齐。这可以看作是没有翻译（t1=···=tM=0）的全局注册问题的一个例子，其中Γ是完整的。不难看出（1.8）可以简化为（1.7）。另一方面，使用第2节中的分析，可以证明在这种情况下（1.6）等同于（1.8）。当Procrustes问题是NP难问题时，已经提出了几个保证多项式时间近似。特别是，在[47，55，46]和最近的[3]中考虑了（1.8）的SDP松弛。我们使用[3]中考虑的（1.7）松弛，原因在第2节中要精确。

**1.5条。符号。**我们用大写字母如O来表示矩阵，用小写字母如t来表示向量。我们用Id表示d×d的单位矩阵，用diag（c1，…，cn）表示n×n的对角矩阵。我们将经常使用由更小的矩阵构建的块矩阵，通常大小为d×d，其中d是环境空间的维数。对于某些块矩阵A，我们将使用Aij来表示它的（i，j）第个块，而A（p，q）来表示它的（p，q）第次项。如果每一块都有

（1≤p，q≤d）。

我们用0表示A是半正定的，也就是说，所有u的uAu T≥0O（d）表示作用于R的正交变换（矩阵）组，用O（d）M表示O（d）的M次积。我们还将方便地识别矩阵[O1····OM]的元素为O（d）M，其中每个Oi∈O*d*（d） 一。我们用来表示x∈的欧几里德范数R（n通常从上下文中可以清楚地看到，如果不是这样，就会指出）。我们用Tr（a）表示方阵a的迹。Frobenius和谱范数定义为*n*

还有。

矩阵A和B之间的Kronecker积用A⊗B表示[24]。全一向量用e表示（从上下文中可以明显看出维数），eNi表示长度为N且在第i个位置为1的全零向量。

**1.6条。组织。**在下一节中，我们将介绍导言中描述的最小二乘配准问题的半定松弛。作为参考，我们还介绍了文献[40,68,25]中已经考虑过的密切相关的光谱弛豫。确切的恢复问题在第3节中讨论，随后在第4节中进行随机试验。谱松弛和半定松弛的稳定性分析在第5节中介绍。数值模拟可以在第6节找到，在第7节讨论某些开放性问题。

**2谱和半定弛豫。**（1.6）的最小化涉及无约束变量（全局坐标和面片平移）和约束变量（正交变换）。首先用未知正交变换求解无约束变量，将其表示为后者的线性组合。这将（1.6）简化为形式（1.7）正交变换上的二次优化问题。

特别是，我们将全局坐标和平移合并为一个矩阵：

（2.1）*.*

同样，我们将正交变换合并为一个矩阵：

（2.2）=[O1···OM]∈R*Od*医学博士*.*

回想一下，我们将方便地将O与O（d）M的元素标识。

为了用Z和O表示（1.6），我们写xk−ti=Zeki，其中

*.*

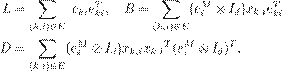
同样，我们写Oi=O（eMi⊗Id）。给我们这个

*.*

使用）和跟踪的属性，我们得到

（2.3）*,*

哪里

（2.4）*,* 和

矩阵L是Γ[14]的组合图Laplacian，其大小为（N+M）×个

（N+M）。矩阵B的大小为Md×（N+M），分块对角矩阵D的大小为Md×Md。

优化程序现在显示

                     （P） 最小φ（Z，O）服从Z∈R∈O（d）M。*d*×（N+M）*，O*

*Z、 O*

O（d）M是非凸的事实使（P）非凸。在2.1-2.5小节中，我们将展示如何用可处理的谱和凸程序来近似这个非凸程序。

**2.1条。优化翻译。**注意，我们可以把（P）写成

*.*

也就是说，对于某个固定的O∈O（d）M，我们首先在自由变量Z上最小化，然后对O最小化。

修正一些任意O∈O（d） M，设ψ（Z）=φ（Z，O）。由（2.3）可知ψ（Z）在Z上是二次的，特别是驻点满足

（2.5）

ψ（Z）的Hessian等于2L，从（2.4）可以清楚地看出0。因此，Z是ψ（Z）的极小值。

如果Γ是连通的，那么e是L[14]的零空间中唯一的向量。设L†是L的Moore-Penrose伪逆，它也是半正定的。可以证实

（2.6）LL†=L†L=英寸+米–（N+M）−1埃。

如果将（2.5）乘以L†，则得到

（2.7），

其中t∈是一些全局平移。相反，如果我们把（2.7）乘以L，并使用eTL=0和Be=0的事实，我们得到（2.5）。因此，（2.5）的每个解都是（2.7）的形式。*d*

将（2.7）代入（2.3），得到

（2.8），

哪里

（2.9）。

注意（2.8）去掉了全局翻译t。这并不奇怪，因为φ对于全局翻译是不变的。此外，请注意，我们还没有对O施加正交约束。由于φ（Z，O）≥0，对于任何Z和O，它必然从（2.8）得到。我们将在下文中看到C的谱如何决定（2.8）的凸松弛的性能。

与刚性理论[26]中的应力概念类似，我们可以将（1.6）视为一对面片之间的“应力”之和，当我们尝试使用刚性变换来注册它们时。特别地，（2.8）中的（i，j）项可以看作是由正交变换产生的（中心）第i个和第j个贴片之间的应力。记住这个类比，我们以后将把C称为贴片应力矩阵。

**2.2条。正交变换优化。**现在的目标是针对正交变换优化（2.8）；也就是说，我们将（P）简化为以下问题：

              （P0）根据（OT O）ii=Id（1≤i≤M）的最小Tr（COT O）。

*O*∈R*d*×*医学博士*

这是一个非凸问题，因为O位于非凸（断开）流形上[1]。我们通常将任何使用流形优化来求解（P0）然后使用（2.7）计算坐标的方法称为“使用流形优化对欧几里德变换进行全局注册”（GRET-MANOPT）。

**2.3。光谱松弛和舍入。**根据（P0）中目标的二次性，可以将其放松为谱问题。更确切地说，考虑域

S={O∈R：O的行是正交的，每行有范数√M}。*d*×医学博士

也就是说，我们不要求O∈S中的d×d块是正交的。相反，我们只要求O的行组成一个正交系统，并且每行都有相同的范数。很明显，S是一个比（P0）中的约束条件所确定的更大的域。我们特别考虑（P0）的以下松弛：

                                                           （P1）最小Tr（COT O）。

*O*∈S

这正是一个谱问题，因为全局极小值是由C的谱分解决定的。更准确地说，设μ1≤······································。定义

（2.10）。

那么

（2.11）Tr（。

由于弛豫，W块不能保证在O（d）中。我们将W的每个d×d块取整为其“最近”的正交矩阵。更确切地说，让

]. 对于每1≤i≤M，我们发现）这样

*.*

如前所述，这有一个封闭形式的解，即U∑vt是的SVD。现在，我们将圆形块放回原位并定义

（2.12）。

在最后一步，在（2.7）之后，我们定义

（2.13）。

Z的前N列被视为重建的全局坐标。

我们将这种谱方法称为“利用谱松弛对欧几里德变换进行全局注册”（GRET-SPEC）。算法1总结了GRET-SPEC的主要步骤。我们注意到Bandeira、Singer和Spielman[4]以及Krishnan等人提出了一种类似的用于角同步的光谱算法。[40]用于初始化流形优化。

现在的问题是如何从GRET-SPEC得到O和Z与原问题有关（P）。由于（P1）是通过放宽（P0）中的块正交性约束得到的，因此很明显，如果W的块是正交的，则O和Z是（P）的解，即所有Z∈的解R∈O（d）M。*d*×（N+M）*，O*

**算法1。**GRET规范。



**要求：**隶属度图，局部坐标{xk，i，（k，i）∈E（Γ）}，维数d。

**确保：**全局坐标x1，…，xN，单位为R。*d*

1： 使用Γ在（2.4）中构建B、L和D。

2： 计算L†和C=D−BL†BT。

三：计算C的底d特征向量，并设置

第四章：   **做**

第六章：

第七章：

9： 返回Z的前N列。



在这种情况下，我们已经找到了原非凸问题（P）的全局极小值。

观察2.1（使用GRET-SPEC进行紧密松弛）。（P1）解的×d块是正交的，然后用GRET-SPEC（P）计算坐标和变换。*如果d是*

如果某些块不是正交的，则四舍五入的量只是（P）解的近似值。

|  |
| --- |
|  |
|  |  |

**2.4条。半定松弛与舍入。**我们现在解释如何使用SDP来获得（P0）的更紧松弛，从而有效地计算全局最小值。我们的SDP是由非凸问题的半定松弛的工作[43，23，60，12]所激励的。考虑域

C={O∈R:（OT O）11=···=（OT O）MM=Id}。*医学博士*×医学博士

也就是说，当我们要求O∈C的每个Md×d块的列正交时，我们不强制非凸秩约束秩（O）=d。这给了我们以下松弛：

（2.14）薄荷糖（COT O）。

*O*∈C

引入变量G=oto，我们看到（2.14）等价于

             （P2）最小Tr（CG）根据。

*G*∈R×Md*医学博士*

这是一个标准的SDP[60]，可以使用SDPT3[58]和CVX[29]等软件包来解决。我们将在后面的2.5小节中提供有关SDP解算器及其计算复杂性的详细信息。

            让我们用，即，

（2.15）Tr（。

根据（P2）中的线性约束，它遵循这个秩（。如果秩（，我们需要用秩-d矩阵取整（近似）。也就是说，我们需要把它投射到（P0）的域上。一种可能是使用带有近似保证的随机舍入；例如，见[23,3]。在这项工作中，我们使用确定性舍入，即特征向量舍入，它保留了顶部的d个特征值而丢弃了剩余的。特别地，设λ1≥λ2≥··≥λMd为G的特征值，q1，…，qMd为相应的特征向量。让

（2.16）。

我们现在继续GRET-SPEC，也就是说，我们从W定义O和Z，如（2.12）和（2.13）。我们将完整的算法称为“使用SDP对欧氏变换进行全局注册”（GRET-SDP）。算法2总结了GRET-SDP的主要步骤。



**算法2。**GRET-SDP。



**要求：**隶属度图，局部坐标{xk，i，（k，i）∈E（Γ）}，维数d。

**确保：**全局坐标x1，…，xN，单位为R。*d*

1： 使用Γ在（2.4）中构建B、L和D。

2： ††BL和L.C。

三：用C语言求解SDP（P2）。

第四章：计算G的上d个特征向量，并设置

5： =1至M do**对于***我*

6： 一。

7： ←。*我我，我*

第八章：

11： 返回Z的前N列。



与观察2.1类似，我们注意到GRET-SDP的以下内容。

观察2.2（使用GRET-SDP进行紧密放松）。（P2）正好是d，然后是GRET-SDP（P）计算的坐标和变换。*如果解的秩是*

如果rank（，则GRET-SDP只能看作（P）解的一个近似值。（P2）近似值的质量可以用例如[3]中的随机取整来量化。更准确地说，因为D是块对角的，（2.14）相当于（直到一个常数项）

最大Tr（QOT O），

*O*∈C

其中0。Bandeira、Kennedy和Singer[3]表明，通过G的某个随机舍入得到的正交变换（我们继续用O表示）满足

选择，

式中，OPT是Q=BL†BT的无松弛问题（1.7）的最优解，αd是具有独立项分布为N（0,1/d）的d×d随机矩阵奇异值的期望平均值。文献[3]推测αd是单调递增的，其边界值为（α1在[48]中也有报道）和α∞=8/3π。关于四舍五入过程的更多细节，以及它与先前工作在近似比方面的关系，我们请读者参考[3]。然而，经验结果表明，就最终重建而言，确定性取整和随机取整之间的差别很小。因此，我们将简单地使用确定性舍入。

**2.5条。计算复杂性。**GRET-SPEC的主要计算是拉普拉斯反演、特征向量计算和正交舍入。当密度为Γ时，转化L的成本为O（（N+M）3）。然而，对于大多数实际应用，我们希望Γ是稀疏的，因为每个点通常都包含在少量的补丁中。在这种情况下，已知线性系统Lx=b可以在时间上以几乎线性的方式在时间上求解，在Γ[57，61]中。应用于（2.6），这意味着我们可以用O（| E（Γ）|）时间（直到对数因子）计算L†。注意，即使L是稠密的，它仍然有可能加速反转

（例如，与直接高斯消去法相比）使用以下公式[33，50]：

*我*†=[L+（N+M）−1 ET]−1−（N+M）−1 ET。

然而，在这种情况下，速度的提高是在绝对运行时间方面。总体复杂度仍然是O（（N+M）3），但常数较小。我们注意到，也可以通过利用Γ[33]的二部性来加速反转，尽管我们在实现中没有使用这种方法。

特征向量计算的复杂度为O（M3d3），而正交取整的复杂度为O（Md3）。例如，使用线性时间拉普拉斯反演，GRET-SPEC的总复杂度为（直到对数因子）

*.*

GRET-SDP中的主要计算块与GRET-SPEC中的相同，包括SDP计算。SDP解可以在多项式时间内使用内点编程计算[67]。特别是，使用内部点解算器（如SDPT3[58]）计算ε-精确解的复杂性为O（（Md）4.5 log（1/ε））。利用（P2）的特殊结构可以降低这种复杂性。例如，注意（P2）中的约束矩阵至多有一个非零系数。使用[30]中的算法，可以将SDP的复杂度降低到O（（Md）3.5log（1/ε））。通过考虑SDP的惩罚版本，我们可以使用一阶解算器（如TFOCS[5]）进一步降低对M和d的依赖性到O（（Md）3ε-1），但代价是对精度的依赖性更强。寻求高效的SDP解算器是当前研究的一个活跃领域。快速SDP解算器已经被提出，它利用SDP解的低秩结构[11，38]或（P2）[64]中线性约束的简单形式。最近，在[22]中提出了一种次线性时间近似算法。因此，使用线性时间拉普拉斯反演和内点SDP解算器的GRET-SDP的复杂度是这样的

*.*

对于SDP变量的大小在150以内的问题，我们可以使用SDPT3[58]或CVX[29]在标准PC上在合理的时间内求解（P2）。在第6节中，我们使用CVX进行数值实验，这些实验涉及小到中等大小的SDP变量。对于较大的SDP变量，可以使用（P2）的低秩结构来加快计算速度。特别是，我们能够使用利用这种低秩结构的SDPLR[11]来求解大小达2000×2000的SDP变量。

**三。准确的恢复。**我们现在研究了隶属度图的条件，在此条件下提出的谱松弛和凸松弛可以从干净的局部坐标（和隶属度图）知识中恢复全局坐标。更精确地说，让´x1，…，x´N是R中点云的真实坐标。假设点云被分成若干块，其成员关系图为Γ，并向我们提供测量值*d*

（3.1）=O´iT（´xk−t´i），（k，i）∈E（Γ），*xk，我*

对于一些O′i∈O（d）和t′i∈R，由Γ和干净测量（3.1）构造了斑片应力矩阵C。问题是，在什么条件下，我们的算法能在什么条件下恢复x´1，…，x´N？我们称之为确切的恢复。要用前面介绍的矩阵表示法表示精确回收率，请定义*d*



和

*O*？=[O´1···O´]∈R×md那么，精确恢复意味着对于某些Ω∈O（d）和t∈R，*米d. d*

（3.2）*.*

从今往后，我们将始终假设Γ是相连的（显然，一个人不可能有准确的恢复）。

Zha和Zhang[68]曾在流形学习中的切线空间对齐的背景下研究过精确恢复的条件，后来Gortler等人也进行了研究。[25]从刚性理论的角度。特别是，他们证明了所谓的仿射刚度概念对于使用谱方法进行精确恢复是足够的。此外，文献[68，25]中的作者将这种刚性概念与其他标准的刚性概念联系起来，并给出了由贴片系统构造的某种超图可以保证仿射刚性的条件。本节的目的是简要介绍[68，25]中的刚度结果，并将这些结果与隶属度图Γ（以及贴片应力矩阵C）的性质联系起来。我们注意到[68，25]中的作者直接检验了全局坐标的唯一性，而我们关心的是通过求解（P1）和（P2）得到的面片变换的唯一性。全局坐标的唯一性是立即的。

命题3.1（唯一性和精确恢复）。（P1）和（P2）有唯一的解决方案，那么（3.2）对GRET-SPEC GRET-SDP都适用*如果和.*

At this point, we note that if a patch has less than d + 1 points, then even when x¯1,...,x¯N are the unique set of coordinates that satisfy (3.1), we cannot guarantee that O¯1,...,O¯M and t¯1,...,t¯M are unique. Therefore, we will work under the mild assumption that each patch has at least d+1 nondegenerate points, so that the patch transforms are uniquely determined from the global coordinates.

We now formally define the notion of affine rigidity. Although phrased differently, it is in fact identical to the definitions in [68, 25]. Henceforth, by affine transform we will mean the group of nonsingular affine maps on R. Affine rigidity is a property of the patch-graph Γ and the local coordinates (xk,i). In keeping with [25], we will collectively call these the patch framework and denote it by Θ = (Γ,(xk,i)).*d*

Definition 3.2 (affine rigidity). ∈ R*Let y*1*,...,yN d be such that, for affine transforms A*1*,...,AM,*

*yk* = Ai(xk,i), (k,i) ∈ E(Γ).

*The patch framework* Θ = (Γ,(xk,i)) is affinely rigid if y1,...,yN is identical to x¯1,...x¯N up to a global affine transform.

Since each patch has d+1 points, we now give a characterization of affine rigidity that will be useful later.

Proposition 3.3. Θ = (Γ,(xk,i)) is affinely rigid if and only if for any F ∈ Rd×Md such that Tr(CFTF) = 0 we must have F = AO¯ for some nonsingular A ∈ Rd×d.*A patch framework*

Before proceeding to the proof, note that O¯ and G¯ = O¯T O¯ are solutions of (P1) and (P2) (this was the basis of Proposition 3.1), and the objective in either case is zero. Indeed, from (3.1), we can write ZL¯ = OB¯ . Since Γ is connected,

(3.3) Z¯ = OBL¯ † + teT (t ∈ R).*d*

Using (3.3), it is not difficult to verify that φ(Z,¯ O¯) = Tr(CG¯). Moreover, it follows from (3.1) that φ(Z,¯ O¯) = 0. Therefore,

(3.4) Tr(CG¯) = Tr(CO¯T O¯) = 0.

Using an identical line of reasoning, we also record another fact. Let F = [F1,...,FM] where each Fi ∈ R. Suppose there exists y1,...,yN ∈ Rand t1,...,tM ∈ Rsuch that*d*×d*d d*

(3.5) yk = Fixk,i + ti, (k,i) ∈ E(Γ).

Then Y = [y1,...,yN,t1,...,tM] ∈ Rsatisfies*d*×(N+M)

(3.6) Y = FBL† + teT

and Tr(CFT F) = 0.

*Proof of Proposition* 3.3. For any F such that Tr(CFT F) = 0, letting

[y1,...,yN,t1,...,tM] = FBL†,

we have (3.5). By the affine rigidity assumption, we must then have yk = Ax¯k +t for some nonsingular A ∈ Rand t ∈ R. Since each patch contains d+1 nondegenerate points, it follows that F = AO¯.*d*×d *d*

In the other direction, assume that y1,...,yN ∈ Rsatisfy (3.5). We know that Tr(CFT F) = 0, and hence F = AO¯ for some nonsingular A. Using (3.6), we immediately have yk = Ax¯k + t. *d* 

Note that Tr(CFT F) = 0 implies that the rows of F are in the null space of C. Therefore, the combined facts that Tr(CFT F) = 0 and F = AO¯ for some nonsingular A ∈ Rd×d are equivalent to saying that null space of C is within the row span of O¯.

The following result then follows as a consequence of (3.3).

Corollary 3.4. Θ = (Γ,(xk,i)) is affinely rigid if and only if the rank of C is (M − 1)d.*A patch framework*

The corollary gives an easy way to check for affine rigidity. However, it is not clear what construction of Γ will ensure such a property. In [68], the notion of graph lateration was introduced that guarantees affine rigidity. Namely, Γ is said to be a graph lateration (or simply laterated) if there exists a reordering of the patch indices such that, for every i ≥ 2, Pi and P1 ∪ ··· ∪ Pi−1 have at least d + 1 nondegenerate nodes in common. An example of a graph lateration is shown in Figure 2.

Theorem 3.5 (see [68]). Γ is laterated and the local coordinates are nondegenerate, then the framework Θ is affinely rigid.*If*

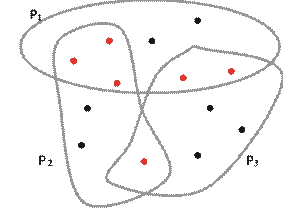


Fig. 2. Instance of three overlapping patches, where the overlapping points are shown in red. In this case, P3 cannot be registered with either P1 or P2 due to insufficient overlap. Therefore, the patches cannot be localized in two dimensions using, for example, the methods in [69, 17] that work by registering pairs of patches. However, the patches can be registered using GRET-SPEC GRET-SDP R*and since the ordered patches P*1*,P*2*,P*3 *form a graph lateration in* 2*.*

Next, we turn to the exact recovery conditions for (P2). The appropriate notion of rigidity in this case is that of universal rigidity [27]. Just as we defined affine rigidity earlier, we can phrase universal rigidity as follows.

Definition 3.6 (universal rigidity). (3.1) holds. Let x1,...,xN ∈ R(s ≥ d) be such that, for some orthogonal Oi ∈ R∈ R*Suppose that ss*×d *and ti s,*

*xk* = Oxik,i + t, i(k,i) ∈ E.

*We say that the patch framework* Θ = (Γ,(xk,i)) is universally rigid in R(xk), we have xk = Ω¯xk for some orthogonal Ω ∈ R*d if for any such s*×d*.*

By orthogonal Ω, we mean that the columns of Ω are orthogonal and of unit norm

(i.e., Ω can be seen as an orthogonal transform in Rby identifying Ras a subspace of R).*s d s*

Following exactly the same arguments used to establish Proposition 3.3, one can derive the following.

Proposition 3.7. *The following statements are equivalent:*

(a) Θ = (Γ,(xk,i)) is universally rigid in R*A patch framework d.*

(b) ∈ R(s ≥ d) be such that OiT Oi = Id for all i. Then*Let O s*×Md

                        Tr(COT O) = 0 ⇒ O = ΩO¯ for some orthogonal Ω ∈ R*s*×d*.*

The question then is, under what conditions is the patch framework universally rigid? This was also addressed in [25] using a graph construction derived from Γ called the body graph. This is given by ΓB = (VB,EB), where VB = {1,2,...,N} and (k,l) ∈ EB if and only if xk and xl belong to the same patch (cf. Figure 3). Next, the following distances are associated with ΓB:

(3.7) , (k,l) ∈ E,B

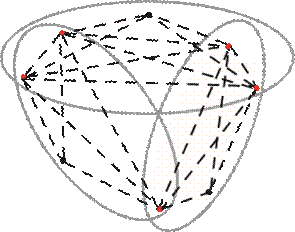


Fig. 3. This shows the body graph for a 3-patch system (patches marked with ovals, points marked with dots). The edges of the body graph are obtained by connecting points that belong to the same patch. The edges within a given patch are marked with the same color. GRET-SDP *can successfully register all the patches if the body graph is rigid in a certain sense.*

where xk,xl ∈ Pi, say. Note that the above assignment is independent of the choice of patch. A set of points (xk)k∈V in Ris said to be a realization of {dkl : (k,l) ∈ E} in R||xk − xl|| ∈ E*s s* if *dkl* = for (*k,l*) .

It was shown in [25] that Θ = (Γ,(xk,i)) is universally rigid if and only if ΓB with distances {dkl : (k,l) ∈ E} has a unique realization in R*s* for all s ≥ d. Moreover, in such a situation, using the distances as the constraints, an SDP relaxation was proposed in [56, 71] for finding the unique realization. We note that although the SDP in [56] has the same condition for exact recovery as (P2), it is computationally more demanding than (P2) since the number of variables is O(N2) for this SDP, instead of O(M2) as in (P2) (for most applications,). Moreover, as we will see shortly in section 6, (P2) also enjoys some stability properties, a fact which has not been established for the SDP in [56].

Finally, we note that universal rigidity is a weaker condition on Γ than affine rigidity.

Theorem 3.8 (see [56, Theorem 2]). *If a patch framework is affinely rigid, then it is universally rigid.*

In [25], it was also shown that the reverse implication is not true using a counterexample for which the patch framework fails to be affinely rigid, but for which the body graph (a Cauchy polygon) has a unique realization in any dimension [15]. This means that GRET-SDP can solve a bigger class of problems than GRET-SPEC, which is perhaps not surprising.

**4. Randomized rank test.** Corollary 3.4 tells us that by checking the rank of the patch-stress matrix C, we can tell whether a patch framework is affinely rigid. In this regard, the patch-stress matrix serves the same purpose as the so-called alignment matrix in [68] and the affinity matrix in [25]. The only difference is that the kernel of C represents the degree of freedom of the affine transform, whereas the kernel of alignment or the affinity matrix directly tells us the degree of freedom of the point coordinates. As suggested in [25], an efficient randomized test for affine rigidity using the concept of affinity matrix can be easily derived. In this section, we describe a randomized test based on the patch-stress matrix, which parallels the proposal in [25]. This procedure is also similar in spirit to the randomized tests for generic local rigidity by Hendrickson [31], for generic global rigidity by Gortler, Healy, and Thurston [26], and for matrix completion by Singer and Cucuringu [54].

Let us continue to denote the patch-stress matrix obtained from Γ and the measurements (3.1) by C. We will use C0 to denote the patch-stress matrix obtained from the same graph Γ, but using the (unknown) original coordinates as measurements, namely,

(4.1) xk,i = ¯xk, (k,i) ∈ Γ.

The advantage of working with C0 over C is that the former can be computed using just the global coordinates, while the latter requires the knowledge of the global coordinates as well as the clean transforms. In particular, this only requires us to simulate the global coordinates. Since the coordinates of points in a given patch are determined up to a rigid transform, we claim the following (cf. section 8.1 for a proof).

Proposition 4.1 (rank equivalence). Γ, C and C0 have the same rank.*For a fixed*

In other words, the rank of C0 can be used to certify exact recovery. The proposed test is based on Proposition 4.1 and the fact that if two different generic configurations are used as input in (4.1) (for the same Γ), then the patch-stress matrices they produce have the same rank. By generic, we mean that the coordinates of the configuration do not satisfy any nontrivial algebraic equation with rational coefficients [26].



**Algorithm 3.** GRET-RRT.



**Require:** Membership graph Γ, and dimension d.

**Ensure:** Exact recovery certificate for GRET-SDP.

1: Build L using Γ, and compute L†.

2: Randomly pick {x1,...,xN} from the unit cube in R, where N = |Vx(Γ)|.*d*

3: for every (k,i) ∈ E(Γ).

*C D BL*†*BT*

5:**then**

6: Positive certificate for GRET-SPEC and GRET-SDP.

7: **else**

8: Negative certificate for GRET-SPEC.

9: GRET-SDP cannot be certified.

10: **end if**



The complete test called “GRET-randomized rank test” (GRET-RRT) is described in Algorithm 3. Note that the main computations in GRET-RRT are the Laplacian inversion (which is also required for the registration algorithm) and the rank computation.

**5. Stability analysis.** We have so far studied the problem of exact recovery from noiseless measurements. In practice, however, the measurements are invariably noisy. This brings us to the question of stability, namely, how stable are GRET-SPEC and GRET-SDP to perturbations in the measurements? Numerical results (to be presented in the next section) show that both the spectral and semidefinite relaxations are quite stable to perturbations. In particular, the reconstruction error degrades quite gracefully with the increase in noise (reconstruction error is the gap between the outputs with clean and noisy measurements). In this section, we try to quantify these empirical observations. In particular, we prove that, for a specific noise model, the reconstruction error grows at most linearly with the level of noise for the semidefinite relaxation.

The noise model we consider is the “bounded” noise model. Namely, we assume that the measurements are obtained through bounded perturbations of the clean measurements in (3.1). More precisely, we suppose that we have a membership graph Γ, and that the observed local coordinates are of the form

(5.1) *,* *.*

In other words, every coordinate measurement is offset within a ball of radius ε around the clean measurements. Here, ε is a measure of the noise level per measurement. In particular, ε = 0 corresponds to the case where we have the clean measurements (3.1).

Since the coordinates of points in a given patch are determined up to a rigid transform, it is clear that the above problem is equivalent to the one where the measurements are

(5.2) *,* *.*

By equivalent, we mean that the reconstruction errors obtained using either (5.1) or

(5.2) are equal. The reason we use the latter measurements is that the analysis in this case is much more simple.

The reconstruction error is defined as follows. Generally, let Z be the output of Algorithms 1 and 2 using (5.2) as input, and let

(5.3) 0 def= [¯x1 ···x¯0···0] ∈ R×(N+M)*ZN d,*

where we assume that the centroid of {x¯1,...,x¯N} is at the origin.

Ideally, we would require that (up to a rigid transformation) when there is no noise, that is, when ε = 0. This is the exact recovery phenomena that we considered earlier. In general, the gap between Z0 and Z is a measure of the reconstruction quality. Therefore, we define the reconstruction error to be

*.*

Note that we are not required to factor out the translation since Z0 is centered by construction.

Our main results are the following.

Theorem 5.1 (stability of GRET-SPEC). {x¯1,...,x¯N}. For fixed noise level ε ≥ 0 and membership graph Γ, suppose we input the noisy measurements (5.2) to*Assume that R is the radius of the smallest Euclidean ball that encloses the clean configuration*

GRET-SPECrank(C0) = (M −1)d, then we have the following bound for GRET-SPEC*. If :*

*,*

*where*



*and*

*.*

*Here λ*2(L) is the second smallest eigenvalue of L.

We assume here that μd+1(C) is nonzero.[3] The bounds here are in fact quite loose. Note that when ε = 0, we recover the exact recovery result for GRET-SPEC provided in [68, 25].

Theorem 5.2 (stability of GRET-SDP). 5.1, we have the following for GRET-SDP*Under the conditions of Theorem :*



The bounds are again quite loose. The main point here is that the reconstruction error for GRET-SDP is within a constant factor of the noise level. In particular, Theorem 5.2 subsumes the exact recovery condition rank(C0) = (M − 1)d described in section

3.

The rest of this section is devoted to the proofs of Theorems 5.1 and 5.2. First, we introduce some notation.

**Notation**. Note that the patch-stress matrix in (P1) is computed from the noisy measurements (5.2), and the same patch-stress matrix is used in (P2). The quantities

, and Z are as defined in Algorithms 1 and 2. We continue to denote the clean patch-stress matrix by C0. Define

*O*0 def= [Id ···Id] and .

Let e1,...,ed be the standard basis vectors of R, and let e be the all-ones vector of length M. Define*d*

(5.4) (1 ≤ i ≤ d).

Note that every d × d block of G0 is Id, and that we can write

(5.5) .

We first present an estimate that applies generally to both algorithms. The proof is provided in subsection 8.2.

Proposition 5.3 (basic estimate). Θ,*Let R be the radius of the smallest Euclidean ball that encloses the clean configuration. Then, for any arbitrary*

*.*

In other words, the reconstruction error in either case is controlled by the rounding error:

(5.7) .

The rest of this section is devoted to obtaining a bound on δ for GRET-SPEC and GRET-SDP.In particular, we will show that δ is of the order of ε in either case. Note that the key difference between the two algorithms arises from the eigenvector rounding, namely, the assignment of the “unrounded” orthogonal transform W (respectively, from the patch-stress matrix and the optimal Gram matrix). However, the analysis in going from W to the rounded orthogonal transform O, and subsequently to Z, is common to both algorithms.

We now bound the error in (5.7) for both algorithms. Note that we can generally write

*,*

where u1,...,ud are orthonormal. In GRET-SPEC, each αi = M, while in GRET-SDP we set αi using the eigenvalues of G.

Our first result gives a control on the quantities obtained using eigenvector rounding in terms of their Gram matrices.

Lemma 5.4 (eigenvector rounding). Θ ∈ O(d) such that*There exist*

*.*

Next, we use a result by Li [42] to get a bound on the error after orthogonal rounding.

Lemma 5.5 (orthogonal rounding). Θ ∈ O(d),*For arbitrary*

*.*

The proofs of Lemma 5.4 and 5.5 are provided in subsections 8.3 and 8.4. At this point, we record a result from [44] which is repeatedly used in the proof of these lemmas and elsewhere.

Lemma 5.6 (see Mirsky [44]). |||·||| be some unitarily invariant norm, and let A,B ∈ R*Let n*×n*. Then*

||| diag(σ1(A) − σ1(B),...,σn(A) − σn(B)) ||| ≤ |||A − B|||.

*In particular, the above result holds for the Frobenius and spectral norms.*

By combining Lemmas 5.4 and 5.5, we have the following bound for (5.7):

(5.8) .

We now bound the quantity on the right in (5.8) for GRET-SPEC and GRET-SDP.

**5.1. Bound for .** GRET-SPECFor the spectral relaxation, this can be done using the Davis–Kahan theorem [7]. Note that from (2.10), we can write

(5.9) .

Following [7, Chap. 7], let A be some symmetric matrix, and let S be some subset of the real line. Denote by PA(S) the orthogonal projection onto the subspace spanned by the eigenvectors of A for which the corresponding eigenvalues are in S. A particular implication of the Davis–Kahan theorem is that

(5.10) ,

where is the complement of S1, and ρ(S1,S2) = min{|u − v| : u ∈ S1,v ∈ S2}.

In order to apply (5.10) to (5.9), set A = C,B = C0,S1 = [μ1(C), μd(C)], and

*S*2 = {0}. If rank(C0) = (M − 1)d, then. Applying (5.10), we get

(5.11) .

Now, it is not difficult to verify that for the noise model (5.2),

*.*

Combining Proposition 5.3 with (5.8), (5.11), and (5.12), we arrive at Theorem 5.1.

**5.2. Bound for .** GRET-SDPTo analyze the bound for GRET-SDP, we require further notation. Recall (5.4), let S be the space spanned by {s1,...,sd} ⊂ R, and let S¯ be the orthogonal complement of S in R. In what follows, we will be required to use matrix spaces arising from tensor products of vector spaces. In particular, given two subspaces U and V of R, denote by U ⊗V the space spanned by the rank-one matrices {uvT : u ∈ U,v ∈ V }. In particular, note that G0 is in S ⊗ S.*MdMdMd*

Let A ∈ Rbe some arbitrary matrix. We can decompose it into*Md*×Md

(5.13) A = P + Q + T,

where



*P* ∈ S ⊗ S, Q ∈ (S ⊗ S) ∪ (S ⊗ S), and T ∈ S ⊗ S.

We record a result about this decomposition from Wang and Singer [63].

Lemma 5.7 (see [63, p. 7]). *Suppose* *and* Δii = 0 (1 ≤ i ≤ M). Let

Δ = P + Q + T as in (5.13). Then

*and* *.*

It is not difficult to verify that Tr(C0G0) = 0 and that 0. From (5.5), we have

*.*

Since each term in the above sum is nonnegative, C0si = 0 for 1 ≤ i ≤ d. In other words, S is contained in the null space of C0. Moreover, if rank(C0) = (M −1)d, then S is exactly the null space of C0. Based on this observation, we give a bound on the residual T.

Proposition 5.8 (bound on the residual). rank(C0) = (M − 1)d. Decompose Δ = P + Q + T as in (5.13). Then*Suppose that*

(5.14) Tr(.

*Proof*. The main idea here is to compare the objective in (P0) with the trace of T. To do so, we introduce the following notation. Let λ1,...,λMd be the full set of eigenvalues of G sorted in nonincreasing order, and let q1,...,qMd be the corresponding eigenvectors. Define

*,*

and define to be the ith Md × d block of O, that is, By construction,. Moreover, by feasibility,

*.*

Thus, the d columns of form an orthonormal system in R. Now define*Md*

*.*

In particular, we will use the fact that () are the minimizers of the unconstrained program

subject to ∈ R(N+M)∈ R*Z Md*×*, O Md*×*Md.*

Note that Tr(). Now, by Lemma (5.7),

Therefore, writing

*,*

we get

Tr(.

*i i*

Therefore,

(5.16) Tr(.

We are done if we can bound the term on the right. To do so, we note from (5.15) that

Tr(.

(k,i)∈E(Γ)

Therefore,

Tr(.

(k,i)∈E(Γ)

To bring in the error term, we write

*,*

and we use ) to get

(5.17) Tr(.

(k,i)∈E

Finally, using the optimality of () for (5.15), we have

*.*

The desired result follows from (5.16), (5.17), and (5.18). 

Finally, we note that Tr(T) can be used to bound the difference between the Gram matrices.

Proposition 5.9 (trace bound). *.*

*Proof*. We will heavily use decomposition (5.13) and its properties. Let

*G*0 + Δ. By the triangle inequality,



*.*

Moreover, since the bottom eigenvalues of G0 are zero, it follows from Lemma 5.6 that the norm of the diagonal matrix is bounded by. Therefore,

(5.19) .

Fix {sd+1,...,sMd} to be some orthonormal basis of S¯. For arbitrary A ∈ R, let*Md*

*A*(p,q) = sAsTp q (1 ≤ p,q ≤ Md).

That is, (A(p,q)) are the coordinates of A in the basis {s1,...,sd}∪ {sd+1,...,sMd}.

Decompose Δ = P + Q + T as in (5.13). Note that P,Q, and T are represented in the above basis as follows: P is supported on the upper d × d diagonal block, T is supported on the lower (M − 1)− 1)d diagonal block, and Q is supported on the off-diagonal blocks. The matrix G is diagonal in this representation.

We can bound using Lemma 5.7,

(5.20) ,

where we have used the properties). In particular,

(5.21) .

On the other hand, since 0, we have (G0 +Δ)(p,q)2 ≤ (G0 +Δ)(p,p)(G0 +

Δ)(q,q). Therefore,

*.*

Notice that 0 = Tr(Δ) = Tr(T) + Tr(P). Therefore,

(5.22) .

Combining (5.19), (5.20), (5.22), and (5.21), we get the desired bound. 

Combining (5.8) with Propositions 5.3, 5.8, and 5.9, we arrive at Theorem 5.2.

**6. Numerical experiments.** We now present some numerical results on multipatch registration using GRET-SPEC and GRET-SDP. In particular, we study the exact recovery and stability properties of the algorithm. We define the reconstruction error in terms of the root-mean-square deviation (RMSD) given by

(6.1) RMSD = min .

In other words, the RMSD is calculated after registering (aligning) the original and the reconstructed configurations. We use the SVD-based algorithm [2] for this purpose.

**Experiment 1**. We first consider a few examples concerning the registration of three patches in R, where we vary Γ by controlling the number of points in the intersection of the patches. We work with the clean data in (3.1) and demonstrate exact recovery (up to numerical precision) for different Γ.2

In the left plot in Figure 4, we consider a patch system with N = 10 points. The points that belong to two or more patches are marked in red, while the rest are marked in black. The patches taken in the order P1,P2,P3 form a lateration in this case. As predicted by Corollary 3.4 and Theorem 3.5, the rank of the patch-stress matrix C0 for this system must be 2(3 − 1) = 4. This is indeed confirmed by our experiment. We expect GRET-SPEC and GRET-SDP to recover the exact configuration. Indeed, we get a very small RMSD of the order of 1e-7 in this case. As shown in the figure, the reconstructed coordinates obtained using GRET-SDP perfectly match the original ones after alignment.

We next consider the example shown in the center plot in Figure 4. The patch system is not laterated in this case, but the rank of C0 is 4. Again we obtain a very small RMSD of the order 1e-7 for this example. This example demonstrates that lateration is not necessary for exact recovery.

In the next example, we show that the condition rank(C0) = (M − 1)d is not necessary for exact recovery using GRET-SDP. To do so, we use the fact that the universal rigidity of the body graph is both necessary and sufficient for exact recovery. Consider the example shown in the right plot in Figure 4. This has barely enough points in the patch intersections to make the body graph universally rigid. Experiments confirm that we have exact recovery in this case. However, it can be shown that rank(C0) < (M − 1)d = 4.

**Experiment 2**. We now consider the structured PACM data in Rshown in Figure 5. There are a total of 799 points in this example that are obtained by sampling the 3-dimensional PACM logo [17, 20]. To begin with, we divide the point cloud into M = 30 disjoint pieces (clusters) as shown in the figure. We augment each cluster into a patch by adding points from neighboring clusters. We ensure that there are sufficient common points in the patch system so that C0 has rank (M −1)d = 87. We generate the measurements using the bounded noise model in (5.2). In particular, we perturb the clean coordinates using uniform noise over the hypercube [−ε,ε]d. For the noiseless setting, the RMSDs obtained using GRET-SPEC and GRET-SDP are 3.3e-113

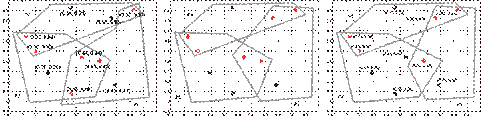


Fig. 4. Instances of a 3-patch system in R4. Right: The body graph is universally rigid but rank(C0) = 3. The original coordinates are marked with ◦, and the coordinates reconstructed by GRET-SDP +.2*. Left: Patch system is laterated. Center: Patch system is not laterated but C*0 *has rank are marked with*

# 0      2            4            6            8            10          12          14          16

Fig. 5. Disjoint clusters for the PACM point cloud. Each cluster is marked with a different color. The clusters are augmented to form overlapping patches which are then registered using GRET-SDP*.*

and 1e-6. The respective RMSDs when ε = 0.5 are 1.4743 and 0.3823. The results are shown in Figure 6.

**Experiment 3**. In the final experiment, we demonstrate the stability of GRET-SDP and GRET-SPEC by plotting the RMSD against the noise level for the PACM data. We use the noise model in (5.2) and vary ε from 0 to 2 in steps of 0.1. For a fixed noise level, we average the RMSD over 20 noise realizations. The results are reported in the bottom plot of Figure 7. We see that the RMSD increases gracefully with the noise level. The result also shows that the semidefinite relaxation is more stable than spectral relaxation, particularly at large noise levels. Also shown in the figure are the RMSDs obtained using GRET-MANOPT with the solutions of GRET-SPEC and GRET-SDP as initialization. In particular, we used the trust region method provided in the Manopt toolbox [10] for solving the manifold optimization (P0). For either initialization, we notice some improvement from the plots. It is clear that the manifold method relies heavily on the initialization, which is not surprising.

Finally, we plot the rank of the SDP solution and notice an interesting phenomenon. Up to a certain noise level, has the desired rank and rounding is not required. This means that the relaxation gap is zero for the semidefinite relaxation, and that we can solve the original nonconvex problem using GRET-SDP up to a certain noise threshold. It is therefore not surprising that the RMSD shows no improvement after we refine the SDP solution using manifold optimization. We have noticed that the rank of the SDP solution is stable with respect to noise for other numerical experiments as well (not reported here).

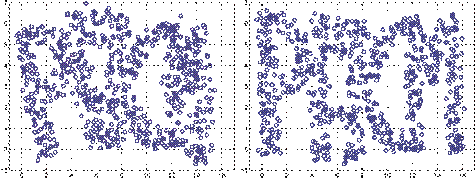


Fig. 6. Reconstruction of the PACM data from corrupted patch coordinates (ε = 0.5). Left: GRET-SPEC1.4743. Right: GRET-SDP= 0.3823. The measurements were generated using the noise model in (5.2).*, RMSD = , RMSD*

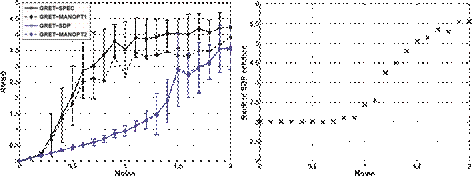


Fig. 7. Left: RMSD versus noise level ε. GRET-MANOPT1 (resp., GRET-MANOPT2) is the result obtained by refining the output of GRET-SPEC GRET-SDP*(resp., ) using manifold optimization. Right: Rank of* *in* GRET-SDP*.*

**7. Discussion.** There are several directions along which the present work could be extended and refined. We summarize some of these below.

1.    *Rank recovery*. Exhaustive numerical simulations (see, for example, Figure 7) show us that the proposed program is quite stable as far as rank recovery is concerned. By rank recovery, we mean that rank(. In this case, the relaxation gap is zero—we have actually solved the original nonconvex problem. We have performed numerical experiments in which we fix some Γ for which rank(C) = (M − 1)d, and gradually increase the noise in the measurements as per the model in (5.2). When the noise is zero, we recover the exact Gram matrix that has rank d. What is interesting is that the program keeps returning a rank-d solution up to a certain noise level. In other words, we observe a phase transition phenomenon in which rank( is consistently d up to a certain noise threshold. This threshold seems to depend on the number of points in the intersection of the patches, which is perhaps not surprising. A precise understanding of this phase transition in terms of the properties of Γ would be an interesting study.

2.    *Conditions on* Γ. We have seen that the universal rigidity of the body graph (derived from Γ) is both necessary and sufficient for exact recovery using GRET-SDP. However, to test unique rigidity, we need to run an SDP [56]. Unfortunately, the complexity of this program is much more than GRET-SDP itself. This led us to consider the rank criteria that could be tested efficiently. The rank test is nonetheless not necessary for exact recovery, and weaker conditions can be found. In particular, an interesting question is whether we could find an efficiently testable condition that would hold for the extreme example in Figure 4, in which Γ fails the rank test.

3.    *Tighter bounds*. The stability in Theorem 5.2 was for the bounded noise model, which made the subsequent analysis quite straightforward. The goal was to establish that the reconstruction error is within Cε for some constant C independent of the noise. In particular, the bounds in Theorem 5.2 are quite loose. One possible direction would be to consider a stochastic noise model with statistically independent perturbations to tighten the bound.

4.    *Anchor points*. In sensor network localization, one has to infer the coordinates of sensors from the knowledge of distances between sensors and their geometric neighbors. In distributed approaches to sensor localization [16, 9], one is faced exactly with the multipatch registration problem described in this paper. Besides the distance information, one often has the added knowledge of the precise positions of selected sensors known as anchors [8]. This is often by design and is used to improve the localization accuracy. The question is whether can we incorporate the anchor constraints into the present registration algorithm. One possible way of leveraging the existing framework is to introduce an additional patch (called anchor patch) for the anchor points. The anchor coordinates are assigned to the points in the anchor patch (treating them as local coordinates). This gives us an augmented bipartite graph Γa, which has one more patch vertex than Γ, and extra edges connecting the anchor patch to the anchor vertices. We then proceed exactly as before; that is, we solve for the global coordinates of both the anchor and nonanchor points given the measurements on Γa.

**8. Technical proofs.** In this section, we give proofs of Propositions 4.1 and 5.3 and Lemmas 5.4 and 5.5.

**8.1. Proof of Proposition 4.1.** We are done if we can show that there exists a bijection between the null space of C and that of C0. To do so, we note that the associated quadratic forms can be expressed as



and

*.*

Here u1,...,uM are the d × 1 blocks of the vector u ∈ R.*Md*×1

Now, it follows from (3.1) that there is a one-to-one correspondence between u

|  |  |
| --- | --- |
| and v, namely, |  |
| *ui* = O¯*ivi* | (1 ≤ i ≤ M), |

such that uTCu = vT C0v. In other words, the null space of C is related to the null space of C0 through an orthogonal transform, as was required to be shown.

**8.2. Proof of Proposition 5.3.** Without loss of generality, we assume that the smallest Euclidean ball that encloses the clean configuration {x¯1,...,x¯N} is centered at the origin, that is,

(8.1) *.*

Let B0 be the matrix B in (2.4) computed from the clean measurements, i.e., from (5.2) with ε = 0. Let B0 + H be the same matrix obtained from (5.2) for some ε > 0. Recall that Z0 = O0B0L† (by the centering assumption in (5.3)). Therefore,

*.*

By the triangle inequality,

(8.2) *.*

Now

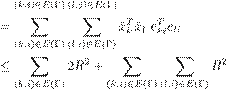
*,*

where λ2(L) is the smallest nonzero eigenvalue of L. On the other hand,

*.*

Using the Cauchy–Schwarz inequality and (8.1), we get

Tr

*.*

Therefore,

(8.3)

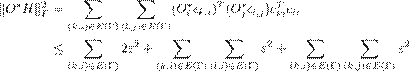
For the other term in (8.2), we can write

*.*

Now

*.*

Therefore, using the Cauchy–Schwarz inequality, the orthonormality of the columns of&apos;s, and the noise model (5.2), we get

*.*

This gives us

(8.4)

Combining (8.2), (8.3), and (8.4), we get the desired estimate.

**8.3. Proof of Lemma 5.4.** The proof is mainly based on the observation that if u and v are unit vectors and 0 ≤ uTv ≤ 1, then

(8.5) .

Indeed,

*.*

To use this result in the present setting, we use the theory of principal angles [7, Chap. 7.1]. This tells us that, for the orthonormal systems {u1,...,ud} and

{s1,...,sd}, we can find Ω1,Ω2 ∈ O(Md) such that

1.    Ω where Θ1 ∈ O(d),

2.    Ω where Θ2 ∈ O(d),

3.    (Ω, and 0 ≤ (Ω1si)T(Ω2ui) ≤ 1 for 1 ≤ i ≤ d.

Here Θ1 and Θ2 are the orthogonal transforms that map {u1,...,ud} and {s1,...,sd} into the corresponding principal vectors.

Using properties 1 and 2 and the fact[4] that αi ≤ M, we can write

*.*

Moreover, by triangle inequality,

*.*

Therefore,

*.*

Now, using (8.5) and the principal angle property 3, we get

*.*

Moreover, using the triangle inequality and properties 1 and 2, we have

*.*



Finally, note that by Lemma 5.6,

(8.6) .

Combining the above relations and setting Θ = Θ, we arrive at Lemma 5.4.

**8.4. Proof of Lemma 5.5.** This is done by adapting the following result by Li [42]: If A,B are square and nonsingular, and if R(A) and R(B) are their orthogonal rounding (obtained from their polar decompositions [32]), then

(8.7) .

We recall that if A = UΣV T is the SVD of A, then R(A) = UV T.

Note that it is possible that some of the blocks of W are singular, for which the above result does not hold. However, the number of such blocks can be controlled by the global error. More precisely, let B ⊂ {1,2,...,M} be the index set such that, for

. Then

*.*

This gives a bound on the size of B. In particular, the rounding error for this set can trivially be bounded as

(8.8) .

On the other hand, we known that, for. From Lemma 5.6, it

follows that



Fix β ≤ 1. Then, and we have from (8.7),

(8.9) .

Fixing β = 1/√2 and combining (8.8) and (8.9), we get the desired bound.

**Acknowledgments.** The authors are grateful to the anonymous referees for their comments and suggestions. They particularly thank the referees for pointing out references [68, 25] and their relation to the present work. The authors also thank Mihai Cucuringu, Lanhui Wang, and Afonso Bandeira for useful discussions, and Nicolas Boumal for advice on the usage of the Manopt toolbox.

REFERENCES

[1] P.-A. Absil, R. Mahony, and R. Sepulchre, Optimization Algorithms on Matrix Manifolds, Princeton University Press, Princeton, NJ, 2009.

[2] K. S. Arun, T. S. Huang, and S. D. Blostein, Least-squares fitting of two 3d point sets, IEEE Trans. Pattern Anal. Mach. Intell., (1987), pp. 698–700.

[3] A. S. Bandeira, C. Kennedy, and A. Singer, Approximating the Little Grothendieck Problem over the Orthogonal Group, preprint, arXiv:1308.5207, 2013.

[4] A. S. Bandeira, A. Singer, and D. A. Spielman, A Cheeger inequality for the graph connection Laplacian, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 1611–1630.

[5] S. R. Becker, E. J. Candes, and M. C. Grant, Templates for convex cone problems with applications to sparse signal recovery, Math. Program. Comput., 3 (2011), pp. 165–218.`

[6] P. J. Besl and N. D. McKay, A method for registration of 3d shapes, IEEE Trans. Pattern Anal. Mach. Intell., 14 (1992), pp. 239–256.

[7] R. Bhatia, Matrix Analysis, Grad. Texts in Math. 169, Springer-Verlag, New York, 1997.

[8] P. Biswas, T.-C. Liang, K.-C. Toh, Y. Ye, and T.-C. Wang, Semidefinite programming approaches for sensor network localization with noisy distance measurements, IEEE Trans. Automat. Sci. Engrg., 3 (2006), pp. 360–371.

[9] P. Biswas, K.-C. Toh, and Y. Ye, A distributed SDP approach for large-scale noisy anchorfree graph realization with applications to molecular conformation, SIAM J. Sci. Comput., 30 (2008), pp. 1251–1277.

[10] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre, Manopt: A MATLAB toolbox for optimization on manifolds, J. Mach. Learn. Res., 15 (2014), pp. 1455–1459.

[11] S. Burer and R. D. C. Monteiro, A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization, Math. Program., 95 (2003), pp. 329–357.

[12] E. Candes and B. Recht, Exact matrix completion via convex optimization, Found. Comput. Math., 9 (2009), pp. 717–772.

[13] E. J. Candes, T. Strohmer, and V. Voroninski, Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming, Comm. Pure Appl. Math., 66 (2013), pp. 1241–1274.

[14] F. R. K. Chung, Spectral Graph Theory, CBMS Regional Conf. Ser. in Math. 92, American Mathematical Society, Providence, RI, 1997.

[15] R. Connelly, Rigidity and energy, Invent. Math., 66 (1982), pp. 11–33.

[16] M. Cucuringu, Y. Lipman, and A. Singer, Sensor network localization by eigenvector synchronization over the Euclidean group, ACM Trans. Sensor Networks, 8 (2012), 19.

[17] M. Cucuringu, A. Singer, and D. Cowburn, Eigenvector synchronization, graph rigidity and the molecule problem, Inform. Inference, 1 (2012), pp. 21–67.

[18] A. Edelman, T. A. Arias, and S. T. Smith, The geometry of algorithms with orthogonality constraints, SIAM J. Matrix Anal. Appl., 20 (1998), pp. 303–353.

[19] K. Fan and A. J. Hoffman, Some metric inequalities in the space of matrices, Proc. Amer. Math. Soc., 6 (1955), pp. 111–116.

[20] X. Fang and K.-C. Toh, Using a distributed SDP approach to solve simulated protein molecular conformation problems, in Distance Geometry, Springer, Berlin, 2013, pp. 351–376.

[21] O. D. Faugeras and M. Hebert, The representation, recognition, and locating of 3d objects, Internat. J. Robotics Res., 5 (1986), pp. 27–52.

[22] D. Garber and E. Hazan, Approximating semidefinite programs in sublinear time, in Advances in Proceedings of Neural Information Processing Systems 2011, Neural Inform. Process. Systems 24, Neural Information Processing Systems (NIPS) Foundation, La Jolla, CA, pp. 1080–1088.

[23] M. X. Goemans and D. P. Williamson, Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming, J. ACM, 42 (1995), pp. 1115– 1145.

[24] G. H. Golub and C. F. Van Loan, Matrix Computations, 3rd ed., Johns Hopkins University Press, Baltimore, 1996.

[25] S. Gortler, C. Gotsman, L. Liu, and D. Thurston, On affine rigidity, J. Comput. Geom., 4 (2013), pp. 160–181.

[26] S. J. Gortler, A. D. Healy, and D. P. Thurston, Characterizing generic global rigidity, Amer. J. Math., 132 (2010), pp. 897–939.

[27] S. J. Gortler and D. P. Thurston, Characterizing the Universal Rigidity of Generic Frameworks, preprint, arXiv:1001.0172, 2009.

[28] J. C. Gower and G. B. Dijksterhuis, Procrustes Problems, vol. 3, Oxford University Press Oxford, UK, 2004.

[29] M. Grant, S. Boyd, and Y. Ye, CVX: MATLAB Software for Disciplined Convex Programming, http://cvxr.com/cvx/. Accessed January 22, 2015.

[30] C. Helmberg, F. Rendl, R. J. Vanderbei, and H. Wolkowicz, An interior-point method for semidefinite programming, SIAM J. Optim., 6 (1996), pp. 342–361.

[31] B. Hendrickson, Conditions for unique graph realizations, SIAM J. Comput., 21 (1992), pp. 65–84.

[32] N. J. Higham, Computing the polar decomposition—with applications, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 1160–1174.

[33] N.-D. Ho and P. Van Dooren, On the pseudo-inverse of the Laplacian of a bipartite graph, Appl. Math. Lett., 18 (2005), pp. 917–922.

[34] B. K. P. Horn, Closed-form solution of absolute orientation using unit quaternions, J. Opt. Soc. Am. A, 4 (1987), pp. 629–642.

[35] S. D. Howard, D. Cochran, W. Moran, and F. R. Cohen, Estimation and Registration on Graphs, preprint, arXiv:1010.2983, 2010.

[36] Q.-X. Huang and L. Guibas, Consistent shape maps via semidefinite programming, in Proceedings of the Eleventh Eurographics/ACMSIGGRAPH Symposium on Geometry Processing (SGP &apos;13), ACM, New York, 2013, pp. 177–186.

[37] A. Javanmard and A. Montanari, Localization from incomplete noisy distance measurements, Found. Comput. Math., 13 (2013), pp. 297–345.

[38] M. Journ´ee, F. Bach, P.-A. Absil, and R. Sepulchre, Low-rank optimization on the cone of positive semidefinite matrices, SIAM J. Optim., 20 (2010), pp. 2327–2351.

[39] J. B. Keller, Closest unitary, orthogonal and Hermitian operators to a given operator, Math. Mag., 48 (1975), pp. 192–197.

[40] S. Krishnan, P. Y. Lee, J. B. Moore, and S. Venkatasubramanian, Global registration of multiple 3d point sets via optimization-on-a-manifold, in Proceedings of the Third Eurographics Symposium on Geometry Processing (SGP &apos;05), ACM, New York, 2005, 187.

[41] G. Lerman, M. McCoy, J. A. Tropp, and T. Zhang, Robust Computation of Linear Models, or How to Find a Needle in a Haystack, preprint, arXiv:1202.4044, 2012.

[42] R.-C. Li, New perturbation bounds for the unitary polar factor, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 327–332.

[43] L. Lovasz and A. Schrijver, Cones of matrices and set-functions and 0 − 1 optimization, SIAM J. Optim., 1 (1991), pp. 166–190.´

[44] L. Mirsky, Symmetric gauge functions and unitarily invariant norms, Quart. J. Math., 11 (1960), pp. 50–59.

[45] N. J. Mitra, N. Gelfand, H. Pottmann, and L. Guibas, Registration of point cloud data from a geometric optimization perspective, in Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing (SGP &apos;04), ACM, New York, 2004, pp. 22–31.

[46] A. Naor, O. Regev, and T. Vidick, Efficient rounding for the noncommutative rothendieck inequality, in Proceedings of the 45th Annual ACM Symposium on Theory of Computing, ACM, New York, 2013, pp. 71–80.

[47] A. Nemirovski, Sums of random symmetric matrices and quadratic optimization under orthogonality constraints, Math. Programming, 109 (2007), pp. 283–317.

[48] Y. Nesterov, Semidefinite relaxation and nonconvex quadratic optimization, Optim. Methods Software, 9 (1998), pp. 141–160.

[49] H. Pottmann, Q.-X. Huang, Y.-L. Yang, and S.-M. Hu, Geometry and convergence analysis of algorithms for registration of 3d shapes, Internat. J. Comput. Vision, 67 (2006), pp. 277– 296.

[50] G. Ranjan, Z.-L. Zhang, and D. Boley, Incremental Computation of Pseudo-inverse of Laplacian: Theory and Applications, preprint, arXiv:1304.2300, 2013.

[51] S. Rusinkiewicz and M. Levoy, Efficient variants of the ICP algorithm, in Proceedings of the Third International Conference on 3-D Digital Imaging and Modeling (Quebec City, QC), IEEE Press, Piscataway, NJ, 2001, pp. 145–152.

[52] G. C. Sharp, S. W. Lee, and D. K. Wehe, Multiview registration of 3d scenes by minimizing error between coordinate frames, in Proceedings of the 7th European Conference on Computer Vision—Part II, Springer-Verlag, Berlin, 2002, pp. 587–597.

[53] A. Singer, Angular synchronization by eigenvectors and semidefinite programming, Appl. Comput. Harmon. Anal., 30 (2011), pp. 20–36.

[54] A. Singer and M. Cucuringu, Uniqueness of low-rank matrix completion by rigidity theory, SIAM J. Matrix Anal. Appl., 31(2010), pp. 1621–1641.

[55] A. M.-C. So, Moment inequalities for sums of random matrices and their applications in optimization, Math. Programming, 130 (2011), pp. 125–151.

[56] A. M.-C. So and Y. Ye, Theory of semidefinite programming for sensor network localization, Math. Programming, 109 (2007), pp. 367–384.

[57] D. A. Spielman and S.-H. Teng, Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems, in Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing, ACM, New York, 2004, pp. 81–90.

[58] K.-C. Toh, M. J. Todd, and R. H. Tutuncu, SDPT3—A MATLAB software package for semidefinite programming, version 1.3, Optim. Methods Software, 11 (1999), pp. 545–581.

[59] T. Tzeneva, Global Alignment of Multiple 3-D Scans Using Eigenvector Synchronization, Senior thesis, Department of Mathematics, Princeton University, Princeton, NJ, 2011.

[60] L. Vandenberghe and S. Boyd, Semidefinite programming, SIAM Rev., 38 (1996), pp. 49–95. [61] N. K. Vishnoi, Lx = b, Found. Trends Theoret. Comput. Sci., 8 (2012), pp. 1–141.

[62] I. Waldspurger, A. d&apos;Aspremont, and S. Mallat, Phase recovery, MaxCut and complex semidefinite programming, Math. Program. Ser. A, 149 (2015), pp. 47–81.

[63] L. Wang and A. Singer, Exact and stable recovery of rotations for robust synchronization, Inform.Inference, 2 (2013), pp. 145–193.

[64] Z. Wen, D. Goldfarb, S. Ma, and K. Scheinberg, Block coordinate descent methods for semidefinite programming, in Handbook on Semidefinite, Conic and Polynomial Optimization, Springer, Berlin, 2012, pp. 533–564.

[65] J. A. Williams and M. Bennamoun, Simultaneous registration of multiple point sets using orthonormal matrices, in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (Istanbul), vol. 4, IEEE Press, Piscataway, NJ, 2000, pp. 2199–2202.

[66] H. Wolkowicz and M. F. Anjos, Semidefinite programming for discrete optimization and matrix completion problems, Discrete Appl. Math., 123 (2002), pp. 513–577.

[67] H. Wolkowicz, R. Saigal, and L. Vandenberghe, eds., Handbook of Semidefinite Programming: Theory, Algorithms, and Applications, Internat. Ser. Oper. Res. Management Sci. 27, Springer, Berlin, 2000.

[68] H. Zha and Z. Zhang, Spectral properties of the alignment matrices in manifold learning, SIAM Rev., 51 (2009), pp. 545–566.

[69] L. Zhang, L. Liu, C. Gotsman, and S. Gortler, An as-rigid-as-possible approach to sensor network localization, ACM Trans. Sensor Networks, 6 (2010), 35.

[70] L. Zhi-Quan, M. Wing-Kin, A.M.-C. So, Y. Ye, and S. Zhang, Semidefinite relaxation of quadratic optimization problems, IEEE Signal Process. Mag., 27 (2010), pp. 20–34.

[71] Z. Zhu, A. M.-C. So, and Y. Ye, Universal rigidity and edge sparsification for sensor network localization, SIAM J. Optim., 20 (2010), pp. 3059–3081.

[[1]](" \l "_ftnref1" \o ") By nondegenerate, we mean that the affine span of the points is d dimensional.

[[2]](" \l "_ftnref2" \o ") The authors thank the anonymous referees for pointing this out.

[[3]](" \l "_ftnref3" \o ") Numerical experiments suggest that this is indeed the case if rank(C0) = (M − 1)d. In fact, we notice a growth in the eigenvalue with the increase in noise level. However, we have not been able to prove this fact.

[[4]](" \l "_ftnref4" \o ") To see why the eigenvalues of G are at most M (the authors thank Afonso Bandeira for suggesting this), note that by the SDP constraints, for every block Gij, we have ) where each xi ∈ R*d*. Then