

随机化算法及其在最小外接圆求解中的应用

石磊, 罗永龙, 张彩云

(安徽师范大学 计算机科学技术系, 安徽 芜湖 241003)

摘要:近年来,随机化算法因其优良的性能吸引了大批学者的关注。在很多问题的求解中,随机化算法常有着惊人的效率,它通常是最快或者是最简单的算法,有时甚至二者兼备。讨论随机化算法在求解最小外接圆中的应用,介绍一个基于随机增量式的递归式算法,对该算法的思想进行详细的叙述,从理论上分析其性能,并结合实验验证,说明该随机化算法具有良好的性能。引入安全性要求,探讨如何在安全多方计算的环境下求解最小外接圆。

关键词:最小外接圆;随机化算法;计算几何;安全多方计算

中图分类号:TP301.6

文献标识码:A

文章编号:1673-629X(2009)08-0082-04

Randomized Algorithm and Its Application in Finding Minimum Circumscribed Circle

SHI Lei, LUO Yong-long, ZHANG Cai-yun

(Department of Computer Science, Anhui Normal University, Wuhu 241003, China)

Abstract: In recent years, randomized algorithm attracts lots of scholars' attention, because of its excellent performance. In many problems, randomized algorithms often have stupendous efficiency, they generally are the fastest or simplest algorithms, even both. Discusses the application of the randomized algorithms in finding the minimal circumscribed circle, introduces a recursive algorithm which is based on the randomized incremental algorithm, and depicts the thinking of the algorithm in detail, proves that it has good performance by analyzing its performance in theory and validating it experimentally. At last, consider the security requirement, research the problem of the minimal circumscribed circle under the environment of secure multi-party computation.

Key words: minimal circumscribed circle; randomized algorithm; computational geometry; secure multi-party computation

0 引言

随着计算机的飞速发展,算法领域的研究也有了很大的进展,人们都在努力追求形式更加简洁、性能更加优良的算法,以提高计算机的工作效率。1970年Berlekamp^[1]提出最早的随机化算法之后,人们便发现了随机化算法的巨大威力,通过很多学者的努力,随机化算法的研究如今已取得了令人瞩目的成绩,它已从单纯的理论研究飞速发展到如今被应用到各类型的算法中^[2~4]。

最小外接圆是计算几何中的一个基本问题。所谓最小外接圆,就是能包含平面中给定点集中的所有点,

且半径最小的那个圆。它的求解在计算机图形学和空间数据库中是非常有用的,在图形学中常用它来建立一个边界,从而来减少不必要的计算量,在空间数据库中,通过最小外接圆来建立索引,也可以大大提高查询速度。由于最小外接圆是包含点集中所有点的最小圆,因此在现实生活中它也有着重要的应用意义,例如根据求解最小外接圆,可以更加合理地确定一些紧急设施的建设位置等。

文献[5~7]研究了圆的一些基本问题,而对于最小外接圆的研究也已经有了很多方法和文献,如最原始的直观穷举法、最优化算法、prune-and-search算法^[8]、基于计算几何知识的求解方法,以及随机增量式算法等。

1 预备知识

本节对文中涉及的一些关键内容进行简要介绍。

定义1(随机化算法) 随机化算法也称概率算法,随机是指在算法中执行某些步骤或某些动作时所进行

收稿日期:2008-12-18;修回日期:2009-03-05

基金项目:国家自然科学基金项目(60703071);安徽省优秀青年科技基金项目(08040106806);安徽省自然科学基金项目(070412043);安徽高校省级自然科学研究重点项目(2006KJ024A)

作者简介:石磊(1986-),女,安徽宁国人,硕士研究生,研究方向为信息安全、分布式计算;罗永龙,博士,教授,研究方向为可信计算、分布式计算、信息安全等。

的选择是随机的。其形式定义一般为:随机化算法是指在算法中使用了随机函数,且随机函数的返回值直接或间接影响了算法的执行流程或执行结果。

定义2 (最小外接圆) 最小外接圆(The Minimum Circumscribed Circle),也可以叫做最小包围圆,最小覆盖圆,是指能够包含平面中所有给定点的最小的圆。用数学语言可以描述为:对于平面上一个包含 n 个点 $P_i(x_i, y_i)$ 的点集 S ,其中 $i = 1, 2, \dots, n$, 寻找一个半径最小的圆 C ,使得 S 中的所有的点要么在 C 中,要么在 C 的边界上。

定义3 (安全多方计算) 安全多方计算(Secure Multi-party Computation, SMC),是指两个或多个用户在不泄漏各自私有的输入信息的条件限制下,共同合作执行一个计算。在计算过程中,参与协作计算的各方,只能了解到自己的输入与最终计算的结果,不能获得其他参与者的任何信息。

2 基于随机化技术的最小外接圆求解算法

迄今为止,最小外接圆的求解已经得到了很多研究,主要如下:

(1)求解最小外接圆的最直观的方法,就是先求出这 n 个点的所有三个点和两个点的组合,再求出所有的每三个点和两个点生成的圆,最后找出最小的且能包含 S 中所有点的那个圆,即为所求的最小外接圆。很显然,求出所有的圆需要 $O(n^3)$ 的时间,而且每个圆都需要花费 $O(n)$ 的时间来检验其是否满足要求,因此该算法总的时间复杂度是 $O(n^4)$;

(2)Elzinga 和 Hearn 在 1972 年提出了一个时间复杂度为 $O(n^2)$ 的算法,该算法的主要思想是减少了需要验证的三元组的数目,使得算法的复杂性减少到 $O(n^2)$;

(3)Shamos 提出了一个时间复杂度为 $O(n \log n)$ 的算法;

(4)1983 年, Nimrod Megiddo 提出了一个 prune-and-search 算法来求解最小外接圆,算法将求解最小外接圆的时间复杂度减少到线性时间 $O(n)$;

(5)在后来的研究中,出现了很多利用计算几何的知识来求解最小外接圆,其中最重要的就是利用到了计算几何中的一个基本知识——凸包,凸包的定义是:给定一个平面上的点集 S ,能够包含 S 中所有点的最小凸集就是凸包。由凸包的定义可以知道,只有位于凸包上的点才有可能构成最小外接圆的点,那么在凸包内部的点对计算最小外接圆是没有任何意义的,因此在计算时就可以删除凸包内部的点以加快计算速度。文献[9]中给出了一个利用凸包求解最小外接圆

的算法。

考虑到随机化算法的优良性能,文中利用随机增量式算法,提出一个带有随机化算法的递归式算法,该算法简洁,并且复杂度也为线性时间。

2.1 算法思想

首先介绍一个引理^[9]:

引理: 设 S 为平面上的一个点集, R 是另一个(允许为空的)点集,而且 $S \cap R = \text{空集}$, 设点 $p \in S$, 则下面命题成立:

(1) 如果存在某个圆覆盖了 S , 而且其边界穿过 R 中的所有点, 那么这样的圆中必然存在唯一的最小者, 将其记作 $mc(S, R)$;

(2) 如果 $p \in mc(S \setminus \{p\}, R)$, 那么 $mc(S, R) = mc(S \setminus \{p\}, R)$;

(3) 如果 $p \notin mc(S \setminus \{p\}, R)$, 那么 $mc(S, R) = mc(S \setminus \{p\}, R \cup \{p\})$ 。

上面的引理文献[9]中给出了详细的说明以及证明。

本节内容就是根据该引理设计一个简单的带有随机化算法的求解包含给定点的最小外接圆算法。将这个引理应用到具体的求解最小外接圆的算法中,可以发现,如果 S 是包含着平面上 n 个点 P_1, P_2, \dots, P_n 的一个点集,其中 S 中点 P 的顺序已经被随机打乱,则可以设计一个算法 $MC(S, R)$ 来求解能包围这 n 个点的最小外接圆,将这 n 个点分为两部分,其中在这个最小外接圆内的点组成的点集就是 S , 而在圆的边界上的那些点则是属于点集 R 。在初始时, S 中即为 P_1, P_2, \dots, P_n , R 为空, 那 $MC(S, R)$ 就是求出能覆盖 S 中 P_1, P_2, \dots, P_n 这 n 个点, 而且边界穿过 R 中所有点的最小外接圆, 而 R 为空, 因此 $MC(S, R)$ 也就是求出能包含给定点集 S 中所有点的最小外接圆。

要想求出包含 P_1, P_2, \dots, P_n 这 n 个点的最小外接圆, 那么就需要先求出包含 P_1, P_2, \dots, P_{n-1} 这 $n-1$ 个点的最小外接圆, 之后判断选取的点 P_n 是否在这个已求的最小外接圆内, 若在, 则该圆即为所求的最小外接圆; 若不在, 则所求的最小外接圆肯定要穿过边界 P_n , 此时就需要求出 $MC(\{P_1, P_2, \dots, P_{n-1}\}, \{P_n\})$, 并且该圆即为所求的最小外接圆。不断地运用递归算法递归调用 $MC(S, R)$ 这个函数, 更新 S 和 R 这两个点集中的点, 直至求出最后的满足要求的最小外接圆即可。

2.2 算法

根据上述思想, 给出算法如下:

首先将输入点集合 $S = \{P_1, P_2, \dots, P_n\}$ 中 P_i 的顺序随机重新排列。

```

Random(A) /* 该算法是将数组中的数排列顺序打乱 */
{ 输入: 含有  $n$  个数  $A[1], A[2], \dots, A[n]$  的数组  $A$ 
  输出: 已经重新排列的数组  $A$ 
  for( $i = n; i \geq 2; i--$ )
  { randomnumber = Random(1,  $i$ ) /* 产生一个 1 到  $i$  之间的随机数 */
    Exchange( $A[i], A[randomnumber]$ ); /* 交换  $A[i]$  和  $A[randomnumber]$  这两个数 */
  }
}

MC(S, R)
{ if( $S$  为空) GetCircle( $R$ ); /*  $C$  为边界穿过  $R$  中所有点的圆 */
  else { 选取点  $P_n$ ;
     $C_1 = MC(S \setminus \{P_n\}, R)$ ;
    if( $P_n \in C_1$ )  $C = C_1$ ;
    else  $C = MC(S \setminus \{P_n\}, R \cup \{P_n\})$ ; /*  $P$  在新圆的边界上 */
  }
}

GetCircle( $R$ ) /* 根据  $R$  中边界点作圆 */
{ if(num = 0)  $C$  为空圆; /* num 为  $R$  中点的个数 */
  else if(num = 1)  $C$  即为该点形成的圆;
  else if(num = 2)  $C$  即为以该两点为直径的圆;
  else  $C$  为边界同时穿过  $R$  中三个点的圆;
}

```

2.3 算法性能分析

2.3.1 理论分析

在这个递归式随机化算法运行初始时, R 为空, S 即为由输入的点所组成的集合。该算法不断递归调用 $MC(S, R)$ 这个函数, 从后向前逐步求解最小外接圆。

这个算法非常简洁, 并且也能在 $O(n)$ 的期望运行时间里完成。

假设算法 $MC(S, R)$ 的期望运行时间是 $T(s, r)$, 其中, s 是点集 S 中的元素个数, r 是点集 R 中的元素个数。下面来分析算法的期望运行时间。

(1) 当 $n = 0$ 时, $T(s, r) = O(1)$;

(2) 当 $n \neq 0$ 时, $T(s, r) = T(s-1, r) + P(s, r) * T(s-1, r-1) + O(1)$ 。

其中 $P(s, r)$ 是随机选取的点 P 不在已求的圆 C 内, 也不在圆 C 边界上, 而是在新圆的边界上的概率。由于 R 中至多只有三个点, 并且已求的圆 C 中已有 r 个点在其边界上, 因此在 s 个点中, 至多只有 $3-r$ 个点是在最小边界圆的边界上。因此, P 点是这样一个点的概率至多为 $(3-r)/n$, 即 $P(s, r) \leq (3-r)/n$, 由此可以计算出 $T(s, r) = O(n)$, 其中 $r = 0, 1, 2, 3$ 。

由理论分析可知, 该改进的递归式算法的时间复杂度为 $O(n)$ 。

同时, 该算法中利用了随机化技术, 因此在对给定点集的最小外接圆的求解中, 算法的运行时间与输入点的顺序是没有任何关系的, 在算法运行中, 是通过随机化的选取点来完成算法的, 因此大大减少了输入等外界因素对算法运行时间影响。

2.3.2 实验验证

在上一节里, 对递归式随机化算法的运行时间进行了理论分析, 为了更好地说明该算法的优良性能, 本节对该算法进行实验验证。实验过程如下:

(1) 输入由平面上的 n 个点组成的集合 S ;

(2) 对集合 S 调用随机化算法, 生成 S 中点的一个随机排列;

(3) 对新的集合 S 执行算法 MC , 不断更新集合 S 和集合 R , 若 S 为空, 则调用算法 $GetCircle$ 生成最小外接圆, 否则每次选取出 S 中的最后一个点 P , 递归调用算法 MC 求出去除 P 点后的那些点的最小外接圆 C , 之后判断 P 是否属于圆 C , 属于则退出该层递归; 否则将 P 加入到边界点的集合 R 中, 继续递归调用算法 MC ;

(4) 不断重复步骤(3), 直至求出这 n 个点的最小外接圆。

根据上述算法以及实验过程, 编写程序测试了运用该算法求解 n 个点的最小外接圆的运算时间(不包含 I/O 时间)。该算法是在 CPU 2.53 GHz, Windows XP 系统, VB 环境中运行, 运行次数为 1000 次。所得实验结果如表 1 所示。

表 1 递归式随机化算法实验测试数据(单位: 秒)

n	20	50	80	100	150	200	500	800	1000	2000
递归增量式 算法时间	0.08	0.29	0.41	0.46	1.04	1.48	2.09	5.07	6.89	7.87

由上表可以看出, 随机递归式的求解算法确实具有很好的性能。

表 2 为对相同的输入实例, 采用普通求解算法与随机化求解算法分别所需要的运行时间的比较(不包含 I/O 时间)。其运行环境为 CPU 2.53GHz, Windows XP 系统, VB 环境中运行, 运行次数为 1000 次。由表 2 可以看出, 在算法中引入随机化技术可以明显改善算法的性能。

表 2 随机化算法时间比较(单位: 秒)

组别	1	2	3	4	5	6	7	8	9	10
普通求解算法	0.22	0.75	0.32	1.39	3.85	7.50	15.8	37.9	63.5	83.2
随机化算法	0.21	1.17	0.12	0.26	1.24	3.16	4.23	10.8	5.72	35.7

3 安全多方环境下的最小外接圆求解

在实际中的应用中, 常出现多方共同合作计算的问题, 而出于安全考虑, 参加计算的各方都希望能保护

自己的私有信息,该问题即为安全多方计算。该问题由 A. C. Yao 首次提出^[10],随后得到了广泛的理论研究^[11~20],其中文献^[11]首次将 SMC 引入计算几何领域,提出保护隐私的计算几何问题(Privacy - Preserving Computational Geometry, PPCG),并给出了一些有用的基本协议。

在安全多方计算环境下的最小外接圆问题可以描述为: Alice 有 m 个点, Bob 有 n 个点,他们想合作计算出包含这 $m + n$ 个点的最小外接圆,但是双方均不愿意向对方泄露自己的数据信息,即在保护各自的私有信息的条件下求解最小外接圆。

由计算几何中凸包的定义可知,只有位于凸包上的点才有可能是构成最小外接圆的点,因此可以先秘密求出这 $m + n$ 个点的凸包,再根据凸包上的点,调用文中的随机化算法求出包含这些点的最小外接圆,即为所求的 $m + n$ 个点的最小外接圆。文献^[13]中,设计了保护私有信息的凸包求解协议,根据该协议, Alice 和 Bob 可以秘密求出包含这 $m + n$ 个点的凸包,然后将组成该凸包的点作为算法 MC 的输入,即可安全地求解出最小外接圆。

Alice 和 Bob 分别先在本地图调用算法 MC 求出各自的最小外接圆, Alice 得到一个圆心为 $O_1(X_1, Y_1)$, 半径为 r_1 的圆, Bob 得到一个圆心为 $O_2(X_2, Y_2)$, 半径为 r_2 的圆,当双方的可信度较高时,可以公布自己的圆心,然后计算出最小外接圆,但是,当双方的信任度较低时,该方案是不安全的。

还可考虑一个近似算法,即求出的最小外接圆并不是精确的,可能存在着一定的误差,但是,其透漏的信息较少,更为安全。Alice 和 Bob 分别先在本地图找出最左、最右、最上、最下的四个点,然后使用秘密比较协议,合作找出 $m + n$ 个点中最左、最右、最上、最下四个点,最后求出这四个点的最小外接圆,即为所求,很显然,该方案所求的最小圆并不精确。

文中仅是将最小外接圆引入到安全多方的计算环境中,上述的求解方案都存在着一定的不足,为了更高效安全地求解保护私有信息的最小外接圆,还需要设计特殊的安全多方计算协议。

4 结束语

讨论了随机化算法在求解最小外接圆中的应用,介绍了一个随机化的递归算法来求解最小外接圆。该算法形式简洁,效率较高,通过理论分析与实验统计,说明了该算法的优良性能。并讨论了安全多方计算环境下的最小外接圆求解问题,提出了几种解决方案。

在以后的工作中,将继续探讨如何设计更加高效

安全的协议来解决保护私有信息条件下的最小外接圆问题。

参考文献:

- [1] Berlekamp E R. Factoring Polynomials Over large Finite Fields [J]. Mathematics of Computation, 1970(24): 713 - 735.
- [2] 王萍, 许海洋. 一种新的随机数组组合发生器的研究[J]. 计算机技术与发展, 2006, 16(4): 79 - 81.
- [3] Rabin M O. Probabilistic Algorithm. Algorithms and Their Complexity: Recent Results and new directions[C]// Traub J F. New York: Academic Press, 1976: 21 - 39.
- [4] Solovay R, Strassen V. A Fast Monte - Carlo Test for Primality[J]. SIAM J. computing, 1977, 6(1): 84 - 85.
- [5] 刘书桂, 杨芳, 陶晋. 计算几何在测量技术中的应用 - 求解最小外接圆[J]. 工程图学学报, 2000(3): 83 - 88.
- [6] 孙燮华. 最佳逼近圆周的 正多边形算法[J]. 微机发展(现更名: 计算机技术与发展), 2001, 11(2): 1 - 2.
- [7] 邱文华, 吴建华, 王平. 一种快速的圆检测方法[J]. 计算机技术与发展, 2007, 17(1): 117 - 118.
- [8] Megiddo N. Linear - time algorithms for linear programming in R^3 and related problems[J]. SIAM Journal on Computing, 1983, 12(4): 759 - 776.
- [9] de Berg M, van Kreveld M, Overmars M, et al. 计算几何 - 算法与应用[M]. 第 2 版. 邓俊辉译. 北京: 清华大学出版社, 2005: 99 - 106.
- [10] Yao A C. Protocols for secure computations[C]// In Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science(FOCS). Chicago, USA: IEEE, 1982: 160 - 164.
- [11] Atallah M J, Du Wenliang. Secure Multi - Party Computational Geometry[C]// In Proceedings of the 7th International Conference Workshop on Algorithms and Data Structures. [s. l.]: Springer Verlag, 2001: 165 - 179.
- [12] 吕品, 孙宝林. 信息安全在数据挖掘中的应用[J]. 微机发展(现更名: 计算机技术与发展), 2005, 15(10): 4 - 5.
- [13] Wang Qi, Luo Yonglong, Huang Liusheng. Privacy - preserving protocols for finding the convex hulls[C]// ARES 2008 - 3rd International Conference on Availability, Security, and Reliability, Proceedings. Barcelona: IEEE, 2008: 727 - 732.
- [14] Goldreich O, Micali S, Wigderson A. How to play any mental game[C]// In Proceedings of the 19th Annual ACM Symposium on Theory of Computing. New York: [s. n.], 1987: 218 - 229.
- [15] Goldwasser S. Multi - party computations: Past and present [C]// In Proceedings of the 16th Annual ACM Symposium on Principles of Distributed Computing. Santa Barbara, CA, USA: [s. n.], 1997: 1 - 6.

体进行两两交叉操作;

步骤 8:(变异算子) 对每个个体按概率进行变异操作;转入步骤 4;

步骤 9:给出最佳的参数,并用其训练数据集以获得最佳模型。

4 数值实验

为了验证文中所提出的算法的有效性,实验数据来源于 UCI 公共数据库^[9]的两个数据集,分别为:ala 数据集和 Wine 数据集。由于 Wine 数据集有三类数据,在本实验中只取对应于标号 1 和标号 2 的两类数据点用于数值试验,从训练集中随机选取 10% 的点作为测试集,其余的点组成训练集;而 WPBC 只有两类数据,也从训练集中随机选取 10% 的点作为测试集;而 ala 数据集也只有两类数据,而且还有另外的测试集,所以直接使用即可。所有数值实验都是在 Intel (R) CPU 3. 0G, 1GM RAM 的兼容机上进行的。

为了测试计算出来的正则参数 C 的性能,将本方法选择出来的 C 和 C 取缺省值 1.0 的结果、还有基于传统遗传算法选择出来的 C 进行了比较,用测试正确率来评价算法的性能。

数值试验结果详见表 1 和表 2。

表 1 三种方法选择的参数 C 的实验结果

数据集	$C = 1.0$	识别率	$C (GA)$	识别率	$C (GGA)$	识别率
ala	1.0	84.05%	3.595	84.37%	3.605	84.39%
WPBC	1.0	80.09%	1.738	81.42%	1.732	81.39%
Wine	1.0	96.84%	0.186	98.46%	0.185	98.45%

表 2 GA 与 GGA 的迭代次数比较

数据集	$C (GA)$	迭代次数	$C (GGA)$	迭代次数
ala	3.595	500	3.605	300
WPBC	1.738	500	1.732	300
Wine	0.186	500	0.185	300

从表 1 可以看出,与一般传统算法相比,采用佳点集遗传算法选择 SVM 参数可以提高数据样本的识别

率,其性能与传统 GA 相当。

从表 2 可以看出,采用佳点集遗传算法选择 SVM 参数比传统 GA 的迭代次数要少很多,因而提高了算法的速度。

5 结束语

文中提出了一种运用佳点集遗传算法对支持向量机的参数进行选择的方法,可以自动地为 SVM 选择合适的参数,省去了人工更改 SVM 参数的麻烦,通过数值实验表明,与传统的 GA 相比,该方法不仅具有收敛速度快、迭代次数少等优点,可以提高算法的速度,而且可以获得比较精确结果,具有较强的泛化能力。这种方法也可以应用于其他类型支持向量机的参数优化,具有一定的推广价值。

参考文献:

- [1] Vapnik V N. The Nature of Statistical Learning Theory[M]. New York:Springer,1995.
- [2] Vapnik V N. Statistical Learning Theory[M]. New York: Wiely,1998.
- [3] Courant R, Hilbert D. Methods of Mathematical Physics[M]. New York: Wildyinterscience,1953.
- [4] 王 睿. 关于支持向量机的参数选择方法分析[J]. 重庆师范大学学报:自然科学版,2007,24(2):36-38.
- [5] 李敏强,寇纪松,林 丹,等. 遗传算法的基本理论与应用[M]. 北京:科学出版社,2002.
- [6] 刘 胜,李妍妍. 自适应 GA-SVM 参数选择算法研究[J]. 哈尔滨工程大学学报,2007,28(4):398-402.
- [7] 杜京义,侯媛彬. 基于遗传算法的支持向量回归机参数选取[J]. 系统工程与电子技术,2006,28(9):1430-1433.
- [8] 张 铃,张 钺. 佳点集遗传算法[J]. 计算机学报,2001,24(9):1-6.
- [9] Blake C, Keogh E, Merz C J. UCI repository of machine learning databases[EB/OL]. University of California, Irvine, Department of Information and Computer Science, URL. 1998. <http://www.ics.uci.edu/mllearn/MLRepository.html>.

(上接第 85 页)

- [16] 罗永龙,黄刘生,徐维江,等. 一个保护私有信息的多边形相交判定协议[J]. 电子学报,2007,35(4):685-691.
- [17] 罗永龙,黄刘生,荆巍巍,等. 一个保护隐私的布尔关联规则挖掘算法[J]. 电子学报,2005,33(5):900-903.
- [18] 罗永龙,黄刘生,荆巍巍,等. 保护私有信息的叉积协议及其应用[J]. 计算机学报,2007,30(2):248-254.
- [19] Luo Yong-Long, Huang Liu-Sheng, Chen Guo-liang, et

al. Privacy-preserving distance measurement and its applications[J]. Chinese Journal of Electronics, 2006, 15(2):237-241.

- [20] Luo Yong-Long, Huang Liu-Sheng, Zhong Hong. Secure Two-Party Point-Circle Inclusion Problem [J]. Journal of Computer Science and Technology, 2007, 23(1):88-91.