Gilbert 算法研究及其改进*

汪卫兵，刘秉瀚

（福州大学 数学与计算机科学学院，福建 福州 350108）

摘 要： Gilbert 算法是求解最接近点对问题的一种算法，广泛应用于碰撞检测、数据分类、运动规划等领域。但是，Gilbert 算法的最大缺点是在很多情况下，当它接近最优解时，收敛速度非常慢。在 Gilbert 算法的基础上提出一个新的迭代策略，可以减少算法的迭代次数，加快收敛速度。实验结果证明，改进后的算法求解速度和收敛速度快。

关键词： Gilbert 算法；NPA 算法；碰撞检测；数据分类

中图分类号： TP391.4 **文献标识码：** A **文章编号：** 1674-7720(2013)14-0065-03

Gilbert algorithm research and its improvement

Wang Weibing, Liu Binghan

(College of Mathematics and Computer Science, Fuzhou University, Fuzhou 110004, China)

Abstract: Gilbert algorithm is an algorithm problem solving nearest point problem, it is widely used in the field of collision detection, data classification, and path planning. However, the major drawback of Gilbert algorithm is that in many cases its convergence rate is very slow as it approaches the optimal solution. This paper proposes a new iteration strategy based on Gilbert algorithm, so that we can reduce the number of iterations of the algorithm to speed up the convergence. The experimental results show that the improved algorithm solves the problem very fast and converges quickly.

Key words: Gilbert algorithm; NPA algorithm; collision detection; data classification

Gilbert 算法[1]是一种求解最接近点对算法，即求解凸包外指定点到一群点集组成的凸包的距离，广泛应用于机器人领域。Gilbert 算法是一种迭代算法，属于梯度下降算法，具有很好的全局收敛性，易被计算机实现，而且适用几何的观点来分析。但是，Gilbert 算法的缺点也很明显，特别是接近最优解时收敛过慢。

针对 Gilbert 算法的改进算法有 MDM 算法[2]、NPA 算法[3]以及 CHANG L[4]等人提出的改进算法。MDM 算法利用具有消极影响的训练样本点改进更新策略，在迭代过程中，一旦发现迭代点的组合中具有消极影响的训练样本点，就直接在迭代点的线性组合中删除或是降低该点对目标函数的影响，并同时使得目标函数边缘下降。MDM 算法解决了 Gilbert 算法接近最优解时收敛过慢的问题，但仍需要多次迭代完成计算。NPA 算法结合了 Gilbert 算法和 MDM 算法的特点，选取三角形区域作为迭代点的搜索范围，扩大了迭代点的搜索范围，实验结果显示，它比 MDM 算法收敛更快，但 NPA 算法仍存在计算量很大的不足。参考文献[4]的研究证明了最接近点对问题的最优解一定出现在凸包顶点中两点组成的边上或者几点确定的面上，根据此结论，最优解一定落在凸包边界上。CHANG L 等人据此提出了一种改进的 Gilbert 算法(下文简称 CQW)，当算法迭代到一定次数时，如果发现算法重复选取几个凸包顶点作为算法迭代选取的顶点，就直接计算凸包外指定点到这几点组成的平面或两点组成的线段的距离，它加快了 Gilbert 的收敛，但需要人工设置迭代次数，迭代次数的选取是一个难点。

针对上述算法计算量大的问题，本文分析了 Gilbert 收敛慢的原因，当新的迭代点越来越趋近于最优解时，它一直在最优解附近徘徊，不能快速到达凸包边界。本文通过实验发现，在迭代的过程中，一旦迭代点出现在凸包的边界上，Gilbert 算法会快速收敛。据此，本文在 Gilbert 算法的基础上提出一种新的迭代策略，迭代过程中将 Gilbert 算法原有的梯度方向上的候选点与凸包边界上的候选点进行比对，选取离凸包外指定点更近的候

* 基金项目：福建省科技计划重点项目(2011Y0040)；福建省自然科学基金项目(2012J01263)

选点作为新的迭代点，这样的迭代机制一有机会即将迭代点拉到凸包的边界上，有效避免了在凸包内部最优解附近不停徘徊迭代的情况发生，减少迭代次数，加快收敛速度。实验结果表明，本文改进后的算法与 NPA 算法相比，计算量小，问题的求解速度更快，与 CQW 算法相比，不需要人工设置迭代次数，更容易求出最优解。

1 Gilbert 算法

1.1 最接近点对问题

给定 m 维空间的两个点集 $X=\{x_i\}_{i=1}^{s1}$ 和 $Y=\{y_i\}_{i=1}^{s2}$，U 和 V 表示两个凸包：

$$U=\{u=\sum_{i=1}^{s1} a_i x_i : \sum_{i=1}^{s1} a_i=1, a_i\geqslant 0\} \tag{1}$$

$$V=\{v=\sum_{i=1}^{s2} a_i y_i : \sum_{i=1}^{s2} a_i=1, a_i\geqslant 0\} \tag{2}$$

最接近点对问题（NPP 问题）的目标就是找到两个凸包间的最接近点对，即：

$$\min_{u\in U, v\in V} ||u-v|| \tag{3}$$

$Z=U-V$ 表示凸包 U 和 V 的明可夫斯基差，凸包 Z 有 $s_1\times s_2$ 个顶点，最接近点对问题转化为在凸包 Z 上找到离凸包外指定点（为方便阐述，凸包外指定点默认为坐标轴原点）最近的点（MNP 问题），即：

$$\min_{z\in Z} ||z|| \tag{4}$$

1.2 算法步骤

Gilbert 算法原本解决的是 MNP 问题，在凸包 U 上找到离原点最近的点。下面给出一些相关的数学定义[5]：

定义 1：设 U 为 R^m 的一个凸包，$y\in U$，称

$$\eta(y)=\max_{x\in U} <x, y> \tag{5}$$

为 U 上的支持函数。$<x, y>$表示求向量 x 和向量 y 的内积运算。

定义 2：设 $S(y)$是 R^m 上的一个函数，称

$$S(y)\in\{x : <x, y>=\eta(y)\}\cap X, y\neq 0 \tag{6}$$

为 R^m 上的关联函数。$S(y)$为 U 的一个顶点。

Gilbert 算法具体步骤如下：

(1)初始化。取 $t=1$，在凸包 U 上任取一点 z_1，$z_1\in U$，设定停止精度 ε。

(2)按式(7)进行梯度下降局部优化迭代更新。

$$z_{t+1}=z_t+\alpha_t(S(-z_t)-z_t) \tag{7}$$

其中 z_{t+1} 为 z_t 和 $S(-z_t)$组成的线段上离原点最近的点，更新力度：

$$\alpha_t=\begin{cases}\dfrac{<z_t, z_t-S(-z_t)>}{||z_t-S(-z_t)||^2}, & z_t-S(-z_t)\neq 0\\ 0, & z_t-S(-z_t)=0\end{cases}$$

(3)如果满足条件$||z_t||^2-||z_{t+1}||^2<\varepsilon$，则算法停止，否则令 $t=t+1$，转步骤(2)。

Gilbert 算法首先任取一个样本点 z_1，通过式(6)确定与 z_1 反向量内积最大的凸包顶点 $S(-z_1)$，判断 $S(-z_1)$是否与 z_1 靠得很近，如果是，则停止；否则，按式(7)进行局部更新，即在 $S(-z_1)$和 z_1 连线上求取离原点最近的点(这个更新操作可以保证 z_{t+1} 的范数严格小于 z_t 的范数)，如此反复，最终获得凸集 U 的最小范数点。

1.3 Gilbert 算法的收敛问题

如图 1 所示，三角形 ABC 为一个平面凸包，O 为原点（即凸包外指定点），选取 P_1 作为算法的初始点，迭代点依次沿着 P_2、P_3、P_4 更新，最终到达离 O 点的最近点 P_n。在 $P_i(i=1, 2, \cdots, n)$迭代的过程中，与向量 $\boldsymbol{P}_i$ 内积最小的顶点一直是 B 点或者 C 点，最终问题的解落在线段 BC 上。Gilbert 算法计算 O 点到三角形 ABC 的求解迭代过程如图 2 所示，Gilbert 算法迭代到一定次数后，迭代点一直在最优解附近徘徊，算法收敛速度非常慢，当设定停止精度 $\varepsilon=10^{-6}$ 时，算法迭代了 489 次。

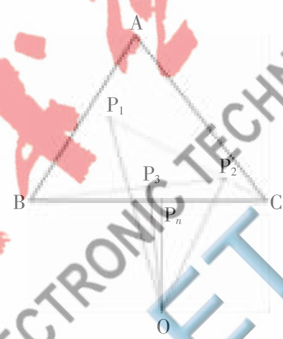

图 1 迭代点依次更新

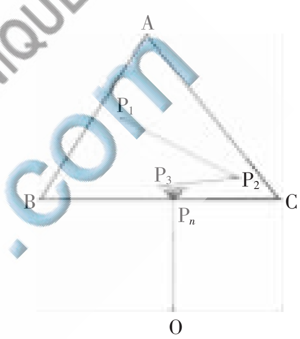

图 2 Gilbert 算法的收敛过程

算法迭代搜索范围如图 3 所示，AA′和 CC′相交于 P_1 点，P_1D 平行于 AC，OE 与 BC 垂直。在第一次迭代过程中，线段 P_1C 是 Gilbert 算法迭代点的搜索范围，Gilbert 算法只计算与向量 $\boldsymbol{P}_1$ 内积最小的顶点 C，而 MDM 算法还计算了与向量 $\boldsymbol{P}_1$ 内积最大的顶点 A，P_1D 平行于 AC，让 P_1 点沿着 P_1D 方向偏移，线段 P_1D 是 MDM 算法迭代点的搜索范围。NPA 算法结合了 Gilbert 算法和 MDM 算法的优点，考虑了 4 种搜索范围：(1)三角形 P_1DC；(2)三角形 P_1A′C；(3)四边形 ACA′C′；(4) 三角形 ABC。这 4 种情况都包含了 Gilbert 算法和 MDM 算法的搜索范围，NPA 算法认为(1)和(2)两种情况优于 Gilbert 算法和 MDM 算法，(3)和(4)两种情况搜索范围太广，比 MDM 算法的效果差。第(1)种情况收敛速度比 MDM 算法快，第(2)种情况比第(1)种情况收敛速度更快。因此，NPA 算法选取三角形 P_1A′C 作为迭代点的搜索区域，扩大了迭代点的搜索范围，有利于找到更优的迭代点，同时加快了收敛速度。CQW 算法依然采用

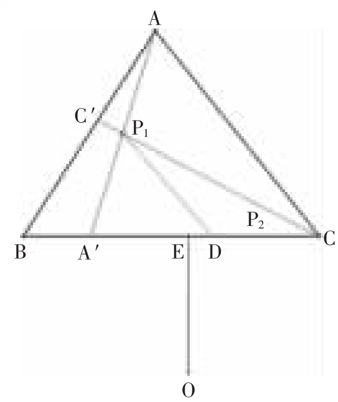

图 3 迭代搜索范围

Gilbert 算法的更新策略，当迭代 K 次（K 的选取是人为设置的）后，发现算法重复选取几个凸包顶点作为算法迭代时选取的顶点，就直接计算 O 点到这几点组成的平面或两点组成的线段的距离，减少了迭代次数。

1.4 Gilbert 算法收敛慢的原因分析

如图 4 所示，矩形 ABCD 是一个平面凸包，O 为原点。利用 Gilbert 算法求解点 O 到凸包的最短距离时，选取初始点 P_1 进行第一次迭代后，与 P_1 点内积最小的顶点为 D 点，然后求出 O 点到线段 P_1D 上的最近点 P_2，并将其作为新的迭代点。图 4 中若 $\angle P_1DO>90°$，则 P_2 点就是 D 点，而 D 属于凸包的边界，算法很快收敛，找到最优解。若 $\angle P_1DO<90°$（如图 5 所示），则迭代点 P_2 出现在凸包的内部，Gilbert 算法会在凸包内部不停地选取迭代点，很难到达凸包的边界，算法收敛速度非常缓慢。

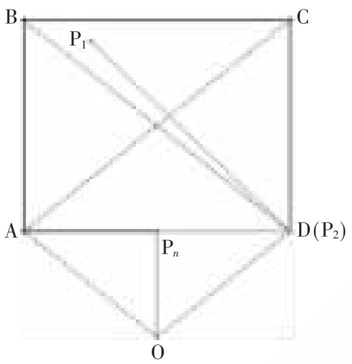

图 4 迭代点出现在凸包边界

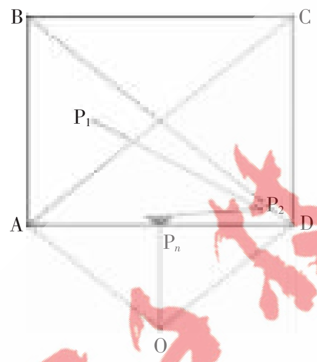

图 5 迭代点在凸包内部

凸包边界由边和面组成，根据 CHANG L[4]等人的结论，最优解一定落在凸包边界上。本文通过多次实验发现，算法迭代点到达凸包边界之后就会快速收敛。因此，本文考虑每次选取迭代点时都与离 O 点最近的凸包边界线段上的点作比较，尽可能将迭代点拉至凸包边界上，这样可避免算法在凸包内部最优解附近徘徊迭代，以此来改进 Gilbert 算法的收敛速度。

2 本文方法

2.1 算法思路

本文基于 1.4 节的分析，在算法第一次迭代确定顶点 D 之后（见图 5），考虑包含 D 的凸包边界线段 AD 和 CD，选择离 O 点最近的边界线段 AD，同时将 P_1D 和 AD 这两条边作为迭代点的搜索范围，分别计算 O 点到这两条线段的最近点，选取两者中较近的点作为新的迭代点：如果 O 点到 AD 边更近，则选取 O 点到 AD 边上最近点作为新的迭代点；如果 O 点到 Gilbert 算法原来的迭代点更近，则选取原来的迭代点作为新的迭代点。这样每次迭代都有机会将迭代点拉到凸包边界，可避免 Gilbert 算法不停地在凸包内部最优解附近选取迭代点，使得算法快速收敛。

2.2 算法步骤

本文改进后的算法步骤如下：

（1）初始化。取 $t=1$，在凸包 U 上任取初始迭代点 z_1，$z_1\in U$，设定停止精度 ε。

（2）按式(7)得到迭代候选点 z'。如果 $z'\in X$（X 为凸包 U 的顶点集合），则 $z_{t+1}=z'$，令 $t=t+1$，转步骤(3)继续执行；如果 $z'\notin X$，按式(8)求 $S(-z')$ 和 $S(-z_t)$ 组成的线段上离 O 点最近的点 z''。

$$z''=S(-z_t)+\beta_t(S(-z')-S(-z_t)) \tag{8}$$

其中：

$$\beta_t=\begin{cases}\dfrac{<S(-z_t),S(-z')-S(-z_t)>}{\|S(-z')-S(-z_t)\|^2}, & S(-z')-S(-z_t)\neq 0\\ 0, & S(-z')-S(-z_t)=0\end{cases}$$

按式(9)选择新的迭代点：

$$z_{t+1}=\min\{\|z'\|,\|z''\|\} \tag{9}$$

令 $t=t+1$，转步骤(3)继续执行。

（3）检验停止条件。如果满足条件 $\|z_t\|^2-\|z_{t+1}\|^2<\varepsilon$，则算法停止，否则转步骤(2)。

如图 6 所示，z_1 是算法选取的初始点，在迭代过程中，先计算出与向量 z_1 内积最小的顶点 C，即 $S(-z_1)$，计算 O 到线段 z_1C 的最近点 z'。

如果 $z'\in X$，则表示新的迭代点是凸包的顶点，按照 Gilbert 原算法执行迭代。如果 $z'\notin X$，计算与向量 z' 内积最小的顶点 B，即 $S(-z')$，计算 O 到 BC 的最近点 z''。

比较 z' 和 z''，取两个距离中较小的垂足作为新一轮的迭代点。这样可以保证每次选取的迭代点到 O 点的距离小于或者等于 Gilbert 算法中的迭代点到 O 点的距离，Gilbert 算法已被证明是收敛的，这样可以保证本文算法的收敛性。

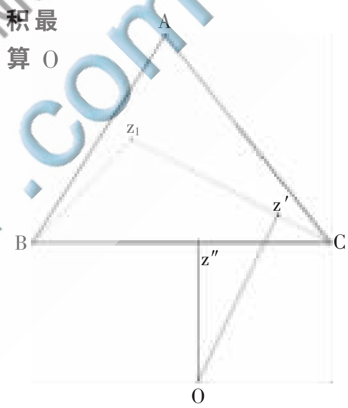

图 6 本文算法迭代过程

本文算法每次选取迭代点时都与离 O 点近的边界线段上的点作比较，有效避免了在凸包内部最优解附近不停地选取迭代点这种情况的发生，可以减少算法的迭代次数，加快收敛速度，提高计算效率。

3 实验结果及分析

为了证明本文算法迭代策略的有效性，将本文算法与 CQW 算法、NPA 算法进行实验对比。设 $X=100\times$rand(50,3)，这是一个 50 乘以 3 的随机数矩阵，它表示 50 个点，每个点的各个坐标值均介于 0~100，U 是由这 50 个顶点构成的凸包，利用上述 3 种算法求解坐标轴原点 O 到凸包的最短距离，设置算法停止精度为 10^{-10}，由于精度较高，Gilbert 算法很难求出最优解，CQW 算法需要人为设定迭代次数，这里设定为 100 次，3 种算法执行时间和迭代次数的比对结果如表 1 所示。求解的最终结果均为 32.813 134 341 830 660。

表 1 实验结果比对

	算法执行时间/s	算法迭代次数
CQW 算法	0.341 474	100
NPA 算法	0.097 737	11
本文算法	0.046 319	11

实验证明，本文算法与 CQW 算法相比，减少了迭代次数，加快了收敛速度；与 NPA 算法相比，提高了计算效率和计算速度。

本文针对 Gilbert 算法的缺点对其进行了改进，解决了 Gilbert 算法收敛过慢的问题，可以非常高效地求解 MNP 问题，改进后的 Gilbert 算法与 Gilbert 算法一样，可用于 NPP 问题，将具有更强的普适性。实验证明，本算法与其他算法相比具有以下优点：(1)算法的迭代次数不需要人为控制，依然可以快速收敛；(2)算法的执行速度较快，最优解的搜索范围比NPA 算法更优；(3)算法非常有效地避免了迭代点在凸包内部不停迭代的情况。改进后的 Gilbert 算法可以用于支持向量机的数据分类、碰撞检测、机器人路径规划等领域。同时，它可以用作支持向量机的训练算法，这是下一步将要展开的工作。

参考文献

[1] GILBERT E G.An iterative procedure for computing the minimum of a quadratic form on a convex set[J].SIAM Journal of Control，1966，4(1)：61-79.

[2] MITCHELL B F，DEM´YANOV V F，MALOZEMOV V N. Finding the point of a polyhedron closest to the origin[J]. SIAM J.Contr.，1974，12：19-26.

[3] KEERTHI S S，SHEVADE S K，BHATTACHARYYA C，et al.A fast iterative nearest point algorithm for support vector machine classifier design[J].IEEE-NN，2000，11(1)：124.

[4] CHANG L，QIAO H，WAN A，et al.An improved Gilbert algorithm with rapid convergence[C].Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems，Beijing：2006：3861-3866.

[5] 周志华.机器学习及其应用 2007[M].北京：清华大学出版社，2007：62-63.

(收稿日期：2013-03-21)

作者简介：

汪卫兵，男，1988 年生，硕士研究生，主要研究方向：模式识别。

刘秉瀚，女，1963 年生，硕士，教授，硕士生导师，主要研究方向：模式识别。