J. Korea Soc. Math. Educ. Ser. B: Pure Appl. Math.
Volume 11, Number 1 (February 2004), Pages 37–49
ISSN 1226-0657

AN ALGORITHM FOR FITTING OF SPHERES

IK-SUNG KIM

ABSTRACT. We are interested in the problem of fitting a sphere to a set of data points in the three dimensional Euclidean space. In Späth [6] a descent algorithm already have been given to find the sphere of best fit in least squares sense of minimizing the orthogonal distances to the given data points. In this paper we present another new algorithm which computes a parametric represented sphere in order to minimize the sum of the squares of the distances to the given points. For any choice of initial approximations our algorithm has the advantage of ensuring convergence to a local minimum. Numerical examples are given.

1. INTRODUCTION

Fitting circles and spheres to given data points in the two or three dimensional Euclidean space is a problem that arises in many application areas. The problem of fitting spheres is at least relevant in computational metrology and reflectometry. Späth [6] generalized a circle fitting algorithm to spheres, which computes a parametric represented sphere in order to minimize the sum of the squares of the distances to the given data points in R^3. In this paper we present another new algorithm which is slightly different from Späth and observe the convergence of our algorithm.

Let us consider a sphere

$$(x-a)^2+(y-b)^2+(z-c)^2=r^2 \tag{1.1}$$

in its parametric form

$$\begin{aligned} x &= a + r\cos u \sin v, \\ y &= b + r\sin u \sin v, \\ z &= c + r\cos v \end{aligned} \tag{1.2}$$

Received by the editors September 2, 2003 and, in revised form, January 27, 2004.
2000 *Mathematics Subject Classification.* 65D10.
Key words and phrases. Orthogonal distance, fitting of spheres, least squares.

For any given set $\{(x_k, y_k, z_k) : k = 1, 2, \ldots, n\}$ of data points in the three dimensional Euclidean space, we are interested in the problem of fitting a sphere to the given data points in such a way that the sum of the squares of the orthogonal distances from (x_k, y_k, z_k) to unknown points $\big(x(a, r; u_k, v_k), y(b, r; u_k, v_k), z(c, r; v_k)\big)$ is minimized. Then the orthogonal distance d_k of a point (x_k, y_k, z_k) can be expressed by

$$(1.3)\quad \begin{aligned} {d_k}^2 &= \min_{u_k, v_k}\Big[\big(x_k - x(a, r; u_k, v_k)\big)^2 + \big(y_k - y(b, r; u_k, v_k)\big)^2 + \big(z_k - z(c, r; v_k)\big)^2\Big] \\ &= \min_{u_k, v_k}\Big[(x_k - a - r\cos u_k \sin v_k)^2 + (y_k - b - r\sin u_k \sin v_k)^2 \\ &\qquad + (z_k - c - r\cos v_k)^2\Big] \end{aligned}$$

and the objective function to be minimized for fitting a sphere to the given n data points (x_k, y_k, z_k) $(k = 1, \ldots, n)$ is given by

$$(1.4)\quad \begin{aligned} Q(\mu) = \sum_{k=1}^{n}\Big[&(x_k - a - r\cos u_k \sin v_k)^2 \\ &+ (y_k - b - r\sin u_k \sin v_k)^2 + (z_k - c - r\cos v_k)^2\Big], \end{aligned}$$

where the parameter vector $\mu = (a, b, c, r, u_1, u_2, \ldots, u_n, v_1, v_2, \ldots, v_n) \in R^{(2n+4)}$.

By using the necessary conditions for a minimum, namely

$$(1.5)\quad \frac{\partial Q}{\partial a} = 0,$$

$$(1.6)\quad \frac{\partial Q}{\partial b} = 0,$$

$$(1.7)\quad \frac{\partial Q}{\partial c} = 0,$$

$$(1.8)\quad \frac{\partial Q}{\partial r} = 0,$$

$$(1.9)\quad \frac{\partial Q}{\partial u_k} = 0 \quad (k = 1, \ldots, n),$$

$$(1.10)\quad \frac{\partial Q}{\partial v_k} = 0 \quad (k = 1, \ldots, n),$$

we have the following $(2n + 4)$ equations:

$$(1.11)\quad \sum_{k=1}^{n} x_k - na - r\sum_{k=1}^{n} \cos u_k \sin v_k = 0,$$

$$\sum_{k=1}^{n} y_k - nb - r\sum_{k=1}^{n} \sin u_k \sin v_k = 0, \tag{1.12}$$

$$\sum_{k=1}^{n} z_k - nc - r\sum_{k=1}^{n} \cos v_k = 0, \tag{1.13}$$

$$\begin{aligned}\sum_{k=1}^{n} x_k \cos u_k \sin v_k + \sum_{k=1}^{n} y_k \sin u_k \sin v_k + \sum_{k=1}^{n} z_k \cos v_k \\ - a\sum_{k=1}^{n} \cos u_k \sin v_k - b\sum_{k=1}^{n} \sin u_k \sin v_k - c\sum_{k=1}^{n} \cos v_k - nr = 0,\end{aligned} \tag{1.14}$$

$$\sin v_k \Big[(x_k - a)\sin u_k - (y_k - b)\cos u_k \Big] = 0 \quad (k = 1, \ldots, n), \tag{1.15}$$

$$\begin{aligned}(x_k - a)\cos u_k \cos v_k + (y_k - b)\sin u_k \cos v_k \\ - (z_k - c)\sin v_k = 0 \quad (k = 1, \ldots, n).\end{aligned} \tag{1.16}$$

The above system of $(2n + 4)$ equations can be divided into two parts. One is a linear system of the four equations (1.11), (1.12), (1.13) and (1.14) for a, b, c and r, whose coefficient matrix depends on u_k and $v_k (k = 1, 2, \ldots, n)$. The other part is concerned with the $2n$ nonlinear equations (1.15) and (1.16). In each of (1.15) and (1.16) the ith equation contains just u_i or v_i as unknown and its coefficients depend on a, b, c and r. There are some well-known methods for solving the linear system of the first part and the $2n$ equations (1.15) and (1.16) can be solved by a simple method. Thus, one may be able to propose mixed iteration algorithms which are connected with both well-known methods above. In this connection, Späth [6] proposed an iteration algorithm for the fitting of spheres.

2. Orthogonal distance fitting of spheres

We will present another algorithm for the fitting of spheres. In fact, the steepest descent method can be employed as a minimization algorithm for minimizing the objective function (1.4). So, the fitting algorithm consists of both the steepest descent method for minimizing $Q(\mu)$ and the root-finding procedure for the $2n$ equations (1.15) and (1.16). Our algorithm for minimizing the given objective function $Q(\mu)$ consists of the following two procedures.

Procedure-1: Solve the $2n$ equations (1.15) and (1.16) with respect to a, b, c and r for u_k and v_k ($k = 1, \ldots, n$). Späth's technique for solving these equations can be seen in Späth [6]. This root-finding procedure is similar to that used in Späth [6].

Case 1: In case of $\sin v_k = 0$ in (1.15), *i. e.*, when $v_k = 0$ or $v_k = \pi$, the equation (1.16) gives $(x_k - a)\cos u_k + (y_k - b)\sin u_k = 0$. If we let

$$\theta^1 = \arctan\left[-\frac{(x_k - a)}{(y_k - b)}\right]\Big(-\frac{\pi}{2} < \theta^1 < \frac{\pi}{2}\Big),$$

then $(\sin u_k, \cos u_k)$ has the two values $(\sin\theta^1, \cos\theta^1)$ and $(-\sin\theta^1, -\cos\theta^1)$. Thus let $u_k = \theta^1$ or $u_k = \theta^1 + \pi$, then $\lambda_k = (u_k, v_k)$ has the following four values:

$$\begin{aligned}\lambda_k^1 &= (\lambda_{k,1}^1, \lambda_{k,2}^1) = (\theta^1, 0),\\ \lambda_k^2 &= (\lambda_{k,1}^2, \lambda_{k,2}^2) = (\theta^1, \pi),\\ \lambda_k^3 &= (\lambda_{k,1}^3, \lambda_{k,2}^3) = (\theta^1 + \pi, 0),\\ \lambda_k^4 &= (\lambda_{k,1}^4, \lambda_{k,2}^4) = (\theta^1 + \pi, \pi).\end{aligned} \tag{2.1}$$

Case 2: If $\sin v_k \neq 0$, then (1.15) gives $(y_k - b)\cos u_k - (x_k - a)\sin u_k = 0$. Let

$$\theta^2 = \arctan\left[\frac{(y_k - b)}{(x_k - a)}\right]\Big(-\frac{\pi}{2} < \theta^1 < \frac{\pi}{2}\Big),$$

then $(\sin u_k, \cos u_k)$ has the two values $(\sin\theta^2, \cos\theta^2)$ and $(-\sin\theta^2, -\cos\theta^2)$. From (1.16) if we let

$$\theta^3 = \arctan\left[\frac{(x_k - a)\cos\theta^2 + (y_k - b)\sin\theta^2}{(z_k - c)}\right],$$

then $(\sin v_k, \cos v_k)$ has the two values $(\sin\theta^3, \cos\theta^3)$ and $(-\sin\theta^3, -\cos\theta^3)$, and the corresponding vector $\lambda_k = (u_k, v_k)$ has the following four values:

$$\begin{aligned}\lambda_k^5 &= (\lambda_{k,1}^5, \lambda_{k,2}^5) = (\theta^2, \theta^3),\\ \lambda_k^6 &= (\lambda_{k,1}^6, \lambda_{k,2}^6) = (\theta^2, \theta^3 + \pi),\\ \lambda_k^7 &= (\lambda_{k,1}^7, \lambda_{k,2}^7) = (\theta^2 + \pi, \theta^3),\\ \lambda_k^8 &= (\lambda_{k,1}^8, \lambda_{k,2}^8) = (\theta^2 + \pi, \theta^3 + \pi).\end{aligned} \tag{2.2}$$

Thus, in any case we can choose $\lambda_k = (u_k, v_k) = \lambda_k^m$ such that

$$\begin{aligned}&\big(x_k - a - r\cos(\lambda_{k,1}^m)\sin(\lambda_{k,2}^m)\big)^2 + \big(y_k - b - r\sin(\lambda_{k,1}^m)\sin(\lambda_{k,2}^m)\big)^2\\ &\quad + \big(z_k - c - r\cos(\lambda_{k,2}^m)\big)^2 = \min_{l=1,2,\ldots,8}\Big[\big(x_k - a - r\cos(\lambda_{k,1}^l)\sin(\lambda_{k,2}^l)\big)^2\\ &\qquad + \big(y_k - b - r\sin(\lambda_{k,1}^l)\sin(\lambda_{k,2}^l)\big)^2 + \big(z_k - c - r\cos(\lambda_{k,2}^l)\big)^2\Big].\end{aligned} \tag{2.3}$$

Further, we get an approximation

$$\begin{aligned}\mu &= (a, b, c, r, u_1, u_2, \dots, u_n, v_1, v_2, \dots, v_n) \\ &= \big(a, b, c, r, \lambda^m_{1,1}, \lambda^m_{2,1}, \dots, \lambda^m_{n,1}, \lambda^m_{1,2}, \lambda^m_{2,2}, \dots, \lambda^m_{n,2}\big).\end{aligned} \tag{2.4}$$

Procedure-2: By using the steepest descent method for the given function $Q(\mu)$, determine a new value

$$\tilde{\mu}^{(j+1)} = \big(\tilde{\mu}_1^{(j+1)}, \tilde{\mu}_2^{(j+1)}, \dots, \tilde{\mu}_s^{(j+1)}\big) = \mu^{(j)} - \alpha \nabla Q(\mu^{(j)})$$

from any given initial approximation $\mu^{(j)} = \big(\mu_1^{(j)}, \mu_2^{(j)}, \dots, \mu_s^{(j)}\big)$. The value of α is given by minimizing the single variable function $h(\alpha) = Q\big(\mu^{(j)} - \alpha \nabla Q(\mu^{(j)})\big)$. Here, ∇ denotes the gradient operator.

We now describe an iteration algorithm for orthogonal distance fitting of spheres.

Algorithm:

Step 0. Let $a^{(0)}$, $b^{(0)}$, $c^{(0)}$ and $r^{(0)}$ be given as initial approximations for unknowns a, b, c and r respectively.

Step 1. Apply **Procedure-1** with $a = a^{(0)}$, $b = b^{(0)}$, $c = c^{(0)}$ and $r = r^{(0)}$ for $u_k = u_k^{(0)}$ and $v_k = v_k^{(0)}$ $(k = 1, \dots, n)$.

If we denote

$$\begin{aligned}\theta^1 &= \arctan\left[-\frac{(x_k - a^{(0)})}{(y_k - b^{(0)})}\right]\Big(-\frac{\pi}{2} < \theta^1 < \frac{\pi}{2}\Big), \\ \theta^2 &= \arctan\left[\frac{(y_k - b^{(0)})}{(x_k - a^{(0)})}\right]\Big(-\frac{\pi}{2} < \theta^1 < \frac{\pi}{2}\Big), \\ \theta^3 &= \arctan\left[\frac{(x_k - a^{(0)})\cos\theta^2 + (y_k - b^{(0)})\sin\theta^2}{(z_k - c^{(0)})}\right],\end{aligned}$$

then $\lambda_k = \lambda_k^{(0)} = (u_k^{(0)}, v_k^{(0)})$ has the following eight values:

$$\begin{aligned}\lambda_k^1 &= (\lambda_{k,1}^1, \lambda_{k,2}^1) = (\theta^1, 0), \\ \lambda_k^2 &= (\lambda_{k,1}^2, \lambda_{k,2}^2) = (\theta^1, \pi), \\ \lambda_k^3 &= (\lambda_{k,1}^3, \lambda_{k,2}^3) = (\theta^1 + \pi, 0), \\ \lambda_k^4 &= (\lambda_{k,1}^4, \lambda_{k,2}^4) = (\theta^1 + \pi, \pi), \\ \lambda_k^5 &= (\lambda_{k,1}^5, \lambda_{k,2}^5) = (\theta^2, \theta^3), \\ \lambda_k^6 &= (\lambda_{k,1}^6, \lambda_{k,2}^6) = (\theta^2, \theta^3 + \pi), \\ \lambda_k^7 &= (\lambda_{k,1}^7, \lambda_{k,2}^7) = (\theta^2 + \pi, \theta^3),\end{aligned} \tag{2.5}$$

$$\lambda_k^8 = (\lambda_{k,1}^8, \lambda_{k,2}^8) = (\theta^2 + \pi, \theta^3 + \pi).$$

Also, we can choose $\lambda_k = (u_k^{(0)}, v_k^{(0)}) = \lambda_k^m$ such that

$$\begin{aligned}(2.6)\quad &\big(x_k - a^{(0)} - r^{(0)}\cos(\lambda_{k,1}^m)\sin(\lambda_{k,2}^m)\big)^2 \\ &+ \big(y_k - b^{(0)} - r^{(0)}\sin(\lambda_{k,1}^m)\sin(\lambda_{k,2}^m)\big)^2 + \big(z_k - c^{(0)} - r^{(0)}\cos(\lambda_{k,2}^m)\big)^2 \\ &= \min_{l=1,2,\dots,8}\Big[\big(x_k - a^{(0)} - r^{(0)}\cos(\lambda_{k,1}^l)\sin(\lambda_{k,2}^l)\big)^2 \\ &+ \big(y_k - b^{(0)} - r^{(0)}\sin(\lambda_{k,1}^l)\sin(\lambda_{k,2}^l)\big)^2 + \big(z_k - c^{(0)} - r^{(0)}\cos(\lambda_{k,2}^l)\big)^2\Big].\end{aligned}$$

Thus we get an approximation

$$\begin{aligned}(2.7)\quad \mu^{(0)} &= \big(a^{(0)}, b^{(0)}, c^{(0)}, r^{(0)}, u_1^{(0)}, u_2^{(0)}, \dots, u_n^{(0)}, v_1^{(0)}, v_2^{(0)}, \dots, v_n^{(0)}\big) \\ &= \big(a^{(0)}, b^{(0)}, c^{(0)}, r^{(0)}, \lambda_{1,1}^m, \lambda_{2,1}^m, \dots, \lambda_{n,1}^m, \lambda_{1,2}^m, \lambda_{2,2}^m, \dots, \lambda_{n,2}^m\big).\end{aligned}$$

Set $j := 0$.

Step 2. Apply **Procedure-2** to the problem of determining an approximation

$$\begin{aligned}&\tilde{\mu}^{(j+1)} \\ &= \big(\tilde{a}^{(j+1)}, \tilde{b}^{(j+1)}, \tilde{c}^{(j+1)}, \tilde{r}^{(j+1)}, \tilde{u}_1^{(j+1)}, \tilde{u}_2^{(j+1)}, \dots, \tilde{u}_n^{(j+1)}, \tilde{v}_1^{(j+1)}, \tilde{v}_2^{(j+1)}, \dots, \tilde{v}_n^{(j+1)}\big)\end{aligned}$$

from $\mu^{(j)} = \big(a^{(j)}, b^{(j)}, c^{(j)}, r^{(j)}, u_1^{(j)}, u_2^{(j)}, \dots, u_n^{(j)}, v_1^{(j)}, v_2^{(j)}, \dots, v_n^{(j)}\big)$.

Let $\nabla Q = \big(\hat{a}^{(j)}, \hat{b}^{(j)}, \hat{c}^{(j)}, \hat{r}^{(j)}, \hat{u}_1^{(j)}, \hat{u}_2^{(j)}, \dots, \hat{u}_n^{(j)}, \hat{v}_1^{(j)}, \hat{v}_2^{(j)}, \dots, \hat{v}_n^{(j)}\big)$ be the gradient of $Q(\mu)$ at $\mu = \mu^{(j)}$ such that

$$\begin{aligned}(2.8)\quad \hat{a}^{(j)} &= -2\Big(\sum_{k=1}^n x_k - na^{(j)} - r^{(j)}\sum_{k=1}^n \cos u_k^{(j)}\sin v_k^{(j)}\Big), \\ \hat{b}^{(j)} &= -2\Big(\sum_{k=1}^n y_k - nb^{(j)} - r^{(j)}\sum_{k=1}^n \sin u_k^{(j)}\sin v_k^{(j)}\Big), \\ \hat{c}^{(j)} &= -2\Big(\sum_{k=1}^n z_k - nc^{(j)} - r^{(j)}\sum_{k=1}^n \cos v_k^{(j)}\Big), \\ \hat{r}^{(j)} &= -2\Big(\sum_{k=1}^n x_k\cos u_k^{(j)}\sin v_k^{(j)} + \sum_{k=1}^n y_k\sin u_k^{(j)}\sin v_k^{(j)} \\ &\quad + \sum_{k=1}^n z_k^{(j)}\cos v_k^{(j)} - a^{(j)}\sum_{k=1}^n \cos u_k^{(j)}\sin v_k^{(j)} \\ &\quad - b^{(j)}\sum_{k=1}^n \sin u_k^{(j)}\sin v_k^{(j)} - c^{(j)}\sum_{k=1}^n \cos v_k^{(j)} - nr^{(j)}\Big),\end{aligned}$$

and for $k = 1, 2, \ldots, n$

$$\hat{u}_k^{(j)} = 2r^{(j)}\Big[\sin v_k^{(j)}\big((x_k - a^{(j)})\sin u_k^{(j)} - (y_k - b^{(j)})\cos u_k^{(j)}\big)\Big],$$

$$\hat{v}_k^{(j)} = -2r^{(j)}\Big[(x_k - a^{(j)})\cos u_k^{(j)}\cos v_k^{(j)} + (y_k - b^{(j)})\sin u_k^{(j)}\cos v_k^{(j)} - (z_k - c^{(j)})\sin v_k^{(j)}\Big].$$

Then it follows from the equations (1.15) and (1.16) that

$$\hat{u}_k^{(j)} = 0 \quad (k = 1, \ldots, n), \qquad \hat{v}_k^{(j)} = 0 \quad (k = 1, \ldots, n). \tag{2.9}$$

We also get an approximation

$$\tilde{\mu}^{(j+1)} = \mu^{(j)} - \alpha\nabla(Q), \tag{2.10}$$

where α is obtained by minimizing the single variable function

$$h(\alpha) = \sum_{k=1}^{n}\Big[\big(x_k - (a^{(j)} - \alpha\hat{a}^{(j)}) - (r^{(j)} - \alpha\hat{r})\cos u_k^{(j)}\sin v_k^{(j)}\big)^2 + \big(y_k - (b^{(j)} - \alpha\hat{b}^{(j)}) - (r^{(j)} - \alpha\hat{r})\sin u_k^{(j)}\sin v_k^{(j)}\big)^2 + \big(z_k - (c^{(j)} - \alpha\hat{c}^{(j)}) - (r^{(j)} - \alpha\hat{r})\cos v_k^{(j)}\big)^2\Big]. \tag{2.11}$$

That is,

$$\alpha = \left(\frac{\sum_{k=1}^{n}(A_k\hat{A}_k + B_k\hat{B}_k + C_k\hat{C}_k)}{\sum_{k=1}^{n}(\hat{A}_k^2 + \hat{B}_k^2 + \hat{C}_k^2)}\right), \tag{2.12}$$

where

$$A_k = a^{(j)} + r^{(j)}\cos u_k^{(j)}\sin v_k^{(j)} - x_k, \quad \hat{A}_k = \hat{a}^{(j)} + \hat{r}^{(j)}\cos u_k^{(j)}\sin v_k^{(j)},$$
$$B_k = b^{(j)} + r^{(j)}\sin u_k^{(j)}\sin v_k^{(j)} - y_k, \quad \hat{B}_k = \hat{b}^{(j)} + \hat{r}^{(j)}\sin u_k^{(j)}\sin v_k^{(j)},$$
$$C_k = c^{(j)} + r^{(j)}\cos v_k^{(j)} - z_k, \quad \hat{C}_k = \hat{c}^{(j)} + \hat{r}^{(j)}\cos v_k^{(j)}.$$

Step 3. Apply **Procedure-1** with $a = \tilde{a}^{(j+1)}$, $b = \tilde{b}^{(j+1)}$, $c = \tilde{c}^{(j+1)}$ and $r = \tilde{r}^{(j+1)}$ for $u_k = u_k^{(j+1)}$ and $v_k = v_k^{(j+1)} (k = 1, \ldots, n)$.

If we let

$$\theta^1 = \arctan\left[-\frac{(x_k - \tilde{a}^{(j+1)})}{(y_k - \tilde{b}^{(j+1)})}\right]\Big(-\frac{\pi}{2} < \theta^1 < \frac{\pi}{2}\Big),$$

$$\theta^2 = \arctan\left[\frac{(y_k - \tilde{b}^{(j+1)})}{(x_k - \tilde{a}^{(j+1)})}\right]\Big(-\frac{\pi}{2} < \theta^1 < \frac{\pi}{2}\Big),$$

$$\theta^3 = \arctan\left[\frac{(x_k - \tilde{a}^{(j+1)})\cos\theta^2 + (y_k - \tilde{b}^{(j+1)})\sin\theta^2}{(z_k - \tilde{c}^{(j+1)})}\right],$$

then $\lambda_k = \lambda_k^{(j+1)} = (u_k^{(j+1)}, v_k^{(j+1)})$ has the following eight values:

$$\begin{aligned}\lambda_k^1 &= (\lambda_{k,1}^1, \lambda_{k,2}^1) = (\theta^1, 0),\\ \lambda_k^2 &= (\lambda_{k,1}^2, \lambda_{k,2}^2) = (\theta^1, \pi),\\ \lambda_k^3 &= (\lambda_{k,1}^3, \lambda_{k,2}^3) = (\theta^1+\pi, 0),\\ \lambda_k^4 &= (\lambda_{k,1}^4, \lambda_{k,2}^4) = (\theta^1+\pi, \pi),\\ \lambda_k^5 &= (\lambda_{k,1}^5, \lambda_{k,2}^5) = (\theta^2, \theta^3),\\ \lambda_k^6 &= (\lambda_{k,1}^6, \lambda_{k,2}^6) = (\theta^2, \theta^3+\pi),\\ \lambda_k^7 &= (\lambda_{k,1}^7, \lambda_{k,2}^7) = (\theta^2+\pi, \theta^3),\\ \lambda_k^8 &= (\lambda_{k,1}^8, \lambda_{k,2}^8) = (\theta^2+\pi, \theta^3+\pi).\end{aligned} \tag{2.13}$$

In addition, we can choose $\lambda_k = \lambda_k^{(j+1)} = (u_k^{(j+1)}, v_k^{(j+1)}) = \lambda_k^m$ such that

$$\begin{aligned}&\big(x_k - \tilde{a}^{(j+1)} - \tilde{r}^{(j+1)}\cos(\lambda_{k,1}^m)\sin(\lambda_{k,2}^m)\big)^2\\ &+ \big(y_k - \tilde{b}^{(j+1)} - \tilde{r}^{(j+1)}\sin(\lambda_{k,1}^m)\sin(\lambda_{k,2}^m)\big)^2\\ &+ \big(z_k - \tilde{c}^{(j+1)} - \tilde{r}^{(j+1)}\cos(\lambda_{k,2}^m)\big)^2\\ = &\min_{l=1,2,\dots,8}\Big[\big(x_k - \tilde{a}^{(j+1)} - \tilde{r}^{(j+1)}\cos(\lambda_{k,1}^l)\sin(\lambda_{k,2}^l)\big)^2\\ &+ \big(y_k - \tilde{b}^{(j+1)} - \tilde{r}^{(j+1)}\sin(\lambda_{k,1}^l)\sin(\lambda_{k,2}^l)\big)^2\\ &+ \big(z_k - \tilde{c}^{(j+1)} - \tilde{r}^{(j+1)}\cos(\lambda_{k,2}^l)\big)^2\Big].\end{aligned} \tag{2.14}$$

Thus we get an approximation

$$\begin{aligned}\mu^{(j+1)} &= \big(a^{(j+1)}, b^{(j+1)}, c^{(j+1)}, r^{(j+1)}, u_1^{(j+1)}, u_2^{(j+1)}, \dots,\\ &\qquad u_n^{(j+1)}, v_1^{(j+1)}, v_2^{(j+1)}, \dots, v_n^{(j+1)}\big)\\ &= \big(a^{(j+1)}, b^{(j+1)}, c^{(j+1)}, r^{(j+1)}, \lambda_{1,1}^m, \lambda_{2,1}^m, \dots, \lambda_{n,1}^m, \lambda_{1,2}^m, \lambda_{2,2}^m, \dots, \lambda_{n,2}^m\big)\end{aligned} \tag{2.15}$$

with $a^{(j+1)} = \tilde{a}^{(j+1)}, b^{(j+1)} = \tilde{b}^{(j+1)}, c^{(j+1)} = \tilde{c}^{(j+1)} r^{(j+1)} = \tilde{r}^{(j+1)}$.

Step 4. If $Q(\mu^{(j+1)}) < Q(\mu^{(j)})$, then we set $j := j+1$ and go back to **Step 2**.

Due to the descent property of **Procedure-2**, one may ensure the convergence of our algorithm to a local minimum independent of approximations $\mu^{(j)}$. That is, it follows

$$Q(\mu^{(j+1)}) \le Q(\mu^{(j)}) \quad (j = 0, 1, 2, \dots). \tag{2.16}$$

Nevertheless, convergence to the global minimum may not be guaranteed. Unfortunately, it is possible for our algorithm to converge to other than the absolute minimum.

3. Numerical examples

To test our algorithm, two examples will be given for the problem of fitting of spheres. By using our algorithm we observe convergence of the corresponding objective function in each case. In fact, it is certain that each quadratic function converges to a local minimum. Further, observing some results obtained by using both our method and Späth's algorithm, each of their convergences will be compared with the other.

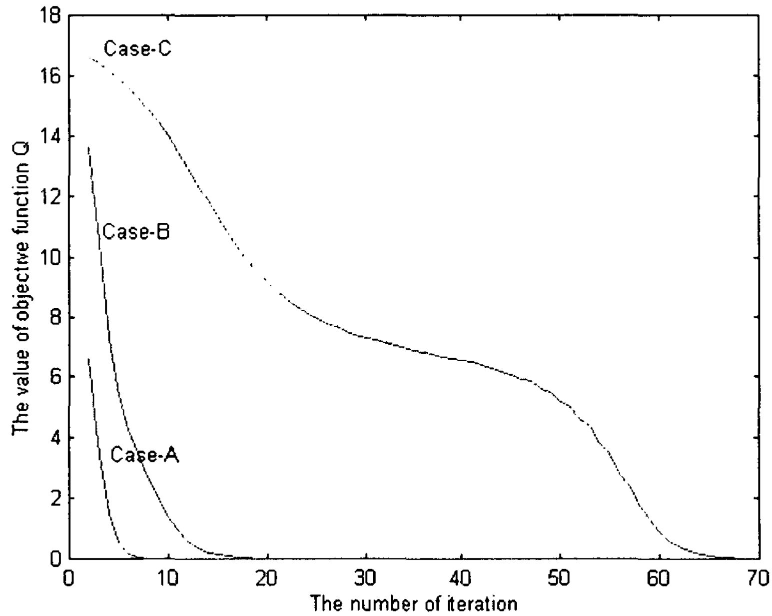

Figure 1. The objective function of our algorithm under different cases

Example 1. The ten data points $(-1,2,3)$, $(0,1,3+\sqrt{2})$, $(0,3,3-\sqrt{2})$, $(1,0,3)$, $(1+\sqrt{2},3,4)$, $(1,2,1)$, $(1,2,5)$, $(1,4,3)$, $(2,1,3-\sqrt{2})$, $(2,2+\sqrt{2},4)$, lying exactly on the sphere $(x-1)^2+(y-2)^2+(z-3)^2=2^2$ with $a=1$, $b=2$, $c=3$ and $r=2$ are given.

When our algorithm or Späth's algorithm is used for determining a sphere $(x - a)^2 + (y - b)^2 + (z - c)^2 = r^2$ best-fitted to the above data points, the objective function to be minimized is given by

$$Q(\mu) = \sum_{k=1}^{10} \left[(x_k - a - r \cos u_k \sin v_k)^2 + (y_k - b - r \sin u_k \sin v_k)^2 + (z_k - c - r \cos v_k)^2 \right].$$

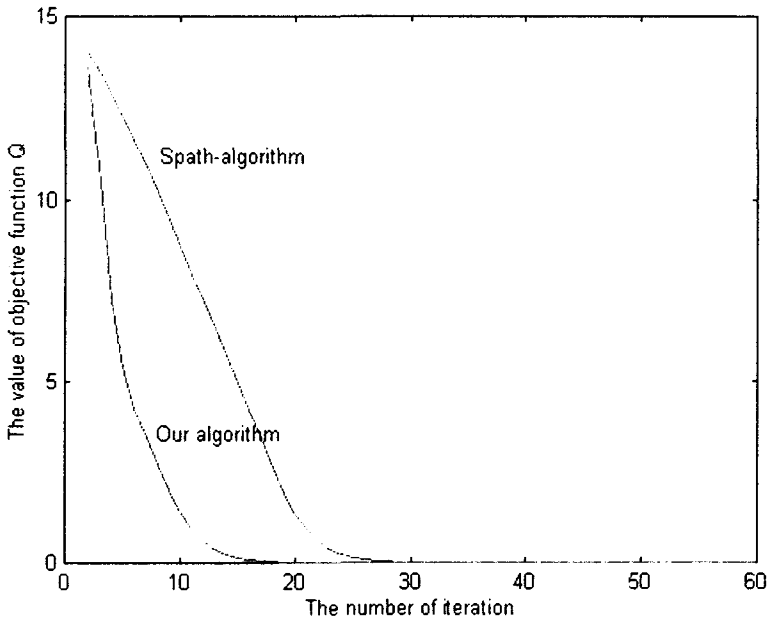

Figure 2. Comparison of our algorithm and Späth's algorithm in terms of convergence

(1) Using **Case-A** ($a^{(0)} = 1$, $b^{(0)} = 1$, $c^{(0)} = 1, r^{(0)} = 1$), **Case-B** ($a^{(0)} = 4$, $b^{(0)} = 4$, $c^{(0)} = 4$, $r^{(0)} = 1$) and **Case-C** ($a^{(0)} = 6$, $b^{(0)} = 6$, $c^{(0)} = 6, r^{(0)} = 1$) for initial approximations in our algorithm respectively, we obtain the exact values $a = 1$, $b = 2$, $c = 3$ and $r = 2$ in less than 100 iterations in each case. We had the exact values after 30 iterations in **Case-A**, after 48 iterations in **Case-B** and after 97 iterations in **Case-C** respectively. These results are visualized in Figure 1. This shows the convergence of $Q(\mu)$ to a local minimum in our algorithm (properly speaking, the global minimum).

(2) Using the same starting points **Case-A** ($a^{(0)} = 1$, $b^{(0)} = 1$, $c^{(0)} = 1$ and $r^{(0)} = 1$) and **Case-B** ($a^{(0)} = 4$, $b^{(0)} = 4$, $c^{(0)} = 4$, $r^{(0)} = 1$) again in Späth's algorithm, we obtained the exact values after 30 iterations in **Case-A** ($a^{(0)} = 1$, $b^{(0)} = 1$,

$c^{(0)} = 1$). This result is almost the same in our algorithm. The good result may be due to the choice of good initial approximations near to the exact solution. On the other hand, in **Case-B** ($a^{(0)} = 4$, $b^{(0)} = 4$, $c^{(0)} = 4$, $r^{(0)} = 1$), we had the exact values after 55 iterations. In Figure 2, we see the convergence of $Q(\mu)$ compared with that in our algorithm.

Example 2. Given any twelve data points $(-3, 2, 3.5)$, $(-2,3,4)$, $(-2,-3,-3)$, $(-1,4,3)$, $(0,3,4)$, $(0,-3,4)$, $(1,4,3)$, $(2,-3,-4)$, $(2,-3,3)$, $(3,-2,3.5)$, $(4,-2,2)$, $(4,-3,0)$ near to the sphere $x^2+y^2+z^2 = 25$, we employ our algorithm and Späth's algorithm by using **Case-A**($a^{(0)} = -7$, $b^{(0)} = -7$, $c^{(0)} = -7$, $r^{(0)} = 1$) and **Case-B**($a^{(0)} = 5$, $b^{(0)} = 5$, $c^{(0)} = 5$, $r^{(0)} = 1$) as initial approximations, respectively.

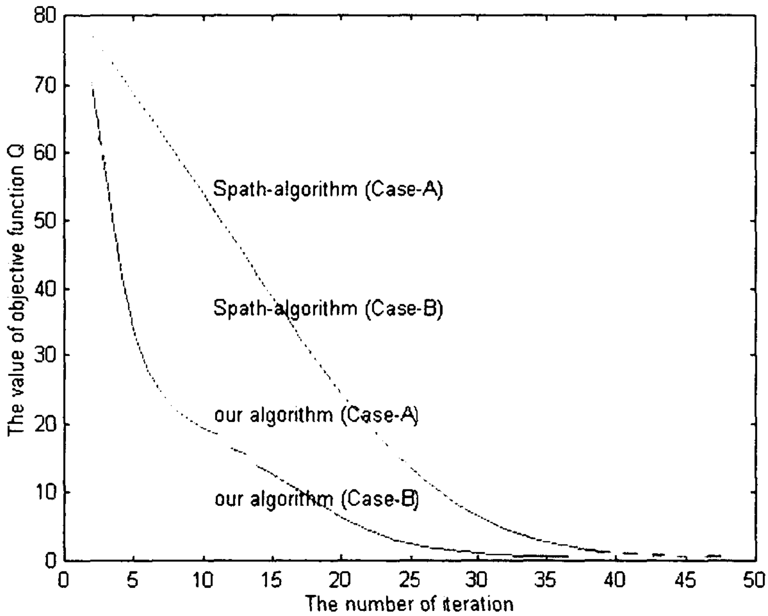

Figure 3. Comparison of our algorithm and Späth's algorithm under different cases

(1) When **Case-A** is used, after 50 iterations, we obtained approximations $a = 0.0222$, $b = 0.1712$, $c = -0.0778$, $r = 5.0628$ and the value of objective function $Q(\mu) = 0.3914$ in our algorithm, and also $a = -0.1194$, $b = 0.0438$, $c = 0.0111$, $r = 5.0403$ and $Q(\mu) = 0.4923$ in Späth's algorithm.
After 100 iterations, we had $a = 0.0881$, $b = 0.2382$, $c = -0.1323$, $r = 5.0844$,

$Q(\mu) = 0.3782$ in our algorithm, and $a = 0.0870$, $b = 0.2373$, $c = -0.1316$, $r = 5.0842$, $Q(\mu) = 0.3782$ in Späth's algorithm.

(2) When **Case-B** is used, after 50 iterations, we got $a = 0.1111$, $b = 0.2635$, $c = -0.1542$, $r = 5.0940$, $Q(\mu) = 0.3800$ and $a = 0.2964$, $b = 0.4339$, $c = -0.2763$, $r = 5.1462$, $Q(\mu) = 0.4869$ in our algorithm and in Späth's algorithm respectively.

(3) In each case of **Case-A** and **Case-B**, the convergences of our algorithm and Späth's algorithm were compared in Figure 3.

(4) Using our algorithm, after 140 iterations in **Case-A** and after 152 iterations in **Case-B**, respectively, we had the optimal approximations $a = 0.0885$, $b = 0.2386$, $c = -0.1327$, $r = 5.0845$ and $Q(\mu) = 0.3782$. These approximations determine a fitted sphere $(x-a)^2 + (y-b)^2 + (z-c)^2 = r^2$. It seems to be the sphere of best fit to the given data points. This shows that the objective function $Q(\mu)$ converges to a local minimum in our algorithm (hopefully, the global minimum). We see this convergence of our algorithm in Figure 4.

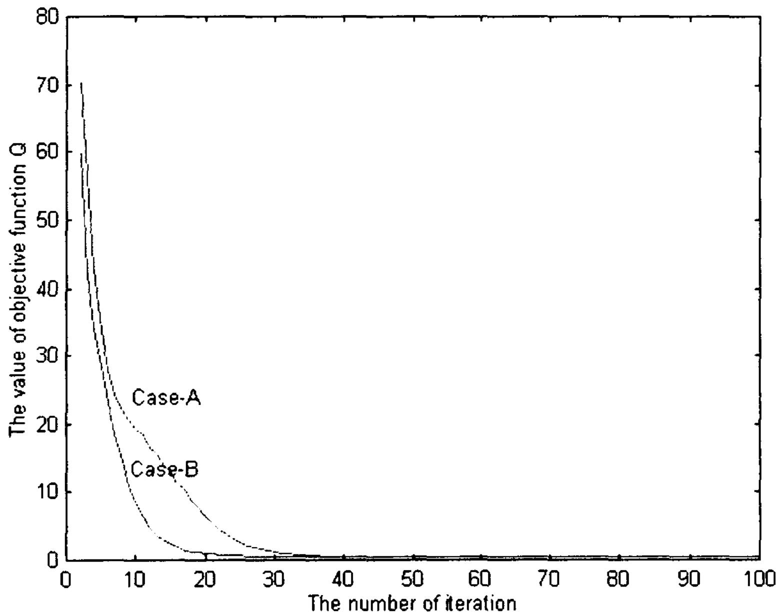

Figure 4. The objective function of our algorithm under different cases

In conclusion, observing the results obtained in the above examples, we can see the following facts which are not proven.

(1) Even if the convergence of the given quadratic function $Q(\mu)$ to the global minimum may not be guaranteed, our algorithm has the advantage of ensuring convergence to a local minimum for any choice of initial approximations.

(2) In comparison with Späth's algorithm by means of the number of iterations without employing computational times, our method is as good as Späth's algorithm in terms of convergence. Further, our algorithm may be better in case that their starting points are not near to the exact solutions.

REFERENCES

1. I. D. Coope: Circle fitting by linear and nonlinear least squares. *J. Optim. Theory Appl.* **76** (1993), no. 2, 381–388. MR **93k**:65054
2. W. Gander, G. H. Golub & R. Strebel: Least-squares fitting of circles and ellipses. *BIT* **34** (1994), no. 4, 558–578. MR **97j**:65020
3. W. Gander & U. von Matt: *Some least squares problems.* In: W. Gander & J. Hřebiček (Eds.), *Solving problems in scientific computing using maple and matlab*, (pp. 251–266). Springer-Verlag, 1993.
4. M. Grossmann: Parametric curve fitting. *Comput. J.* **14** (1971), 169–172. MR **43**#4220
5. H. Späth: Orthogonal squared distance fitting with parabolas. In: Gotz Alefeld & Jurgen Herzberger (Eds.), *Numerical methods and error bounds (Oldenburg, 1995)*, (pp. 261–269), Math. Res., 89, Akademie Verlag, Berlin, 1996. MR **98c**:65016
6. ———: Least-square fitting with spheres. *J. Optim. Theory Appl.* **96** (1998), no. 1, 191–199. MR **98m**:65065

DEPARTMENT OF APPLIED MATHEMATICS, KOREA MARITIME UNIVERSITY, 1 DONGSAM-DONG, YEONGDO-GU, BUSAN 606-791, KOREA
Email address: ikim@mail.hhu.ac.kr