

Bursa 非线性转换模型 7 参数的粒子群解法^{*}

杨 星 朱大栋 赵 钢 李志清
(江苏省水利科学研究院,南京 210017)

摘 要 针对 Bursa 非线性转换模型 7 参数求解中广泛使用的最小二乘法存在的非线性方程矩阵病态问题、初值选取困难问题以及求解收敛性等问题,提出了一种 Bursa 非线性转换模型 7 参数求解的粒子群算法。实例证明:该算法计算结果精确度高,且适合于大旋转角。
关键词 Bursa 模型;7 参数;非线性转换;粒子群算法;空间坐标
中图分类号:P207 **文献标识码**:A

RESEARCH ON BURSA NONLINEAR 7-PARAMETER
TRANSFORMATION WITH PSO ALGORITHM

Yang Xing, Zhu Dadong,Zhao Gang and Li Zhiqing
(Jiangsu Hydraulic Research Institute,Nanjing 210017)

Abstract The 7-parameter is the main parameter for space coordinate transformation,and the minor error of 7-parameter could lead to large coordinate conversion deviances. In order to improve the computational accuracy of 7-parameter,a mathematical model of Bursa 7-parameter analysis based on the partical swarm optimization(PSO) algorithm was established. Compared with a least square method,the model can avoid effectively the matrix's morbid-ity problem,difficulty of selecting initial values,the convergence of algorithm. Finally,the practical application shows that the model has many advantages such as high accuracy and can be adapted large rotation angle.
Key words: Bursa;7-parameter;nonlinear transformation;PSO(Partical Swarn Oplimization);3D coordinate

1 前言

我国工程应用中采用的坐标系统繁杂,有国家坐标系,也有地方独立坐标系,还存在 WGS84 坐标,常需要对这些坐标进行互相转换。数学上,将已知点从一个坐标系统映射到另外一个坐标系统时,若两坐标系统原点不一致,则该点应该首先进行坐标平移,若坐标轴间互不平行,再进行坐标旋转,最后如果两个坐标系尺度不一样,还要进行一次比例缩放,才能获得其在另一个坐标系统下的对应坐标,两

坐标是一种非线性的映射关系。特殊情况下,若两坐标之间不需要旋转变换时,映射是一种线性关系。具体到三维空间坐标系统(X,Y,Z)之间的转换,则存在 3 个平移参数($\Delta X,\Delta Y,\Delta Z$),3 个旋转角度参数($\varepsilon_x,\varepsilon_y,\varepsilon_z$),以及一个尺度变化参数(k),共是 7 个参数,7 参数法从数学角度来说也是最严密的转换方法。目前常用的非线性空间坐标转换模型,如布尔莎(Bursa)模型、莫洛金斯基模型(Molodensky)和武测模型,都采用 7 参数的计算方法^[1],只是在 7 参数的定义上略有差异,转换结果是等价的。

^{*} 收稿日期:2012-04-12
基金项目:水利部公益性项目(201001070)
作者简介:杨星,男,1978 年生,博士,主要从事城市水环境、河流和海岸水动力学、水利工程测量等方面的研究。E-mail: ydaxue@163.com

具体到实际工作中,难点不在于模型的选择,而是转换参数的解算方法,原因是求 7 个转换参数至少需要 3 个及以上的公共点,独立方程数目多于变量数目,不可避免地要遇到非线性超定方程组的求解。一般情况下,超定方程组没有精确解,测量上普遍采用最小二乘法将超定方程组转换成非线性方程组,然后通过 Gauss-Newton 法、Euler 法、ABS 投影法、拟 Newton 法、Brown 法或者割线法等求近似解。文献[2,3]都是基于最小二乘法原理,通过 Gauss-Newton 法求解 7 参数的近似值。由于公共点距离一般较近,非线性方程矩阵病态问题严重,会影响解的准确性^[4]。另外,大量的三角函数不仅影响求解速度、增大计算误差,还存在求解过程中初值选取困难、影响求解收敛性等问题。为此,文献[5,6]基于最小二乘法的同时,辅助采用稳健估计法提高解的精度。文献[7,8]则从解析变换法的角度进行了一些积极的探索。实际上,非线性分析十分繁琐,很难找到一种通用解法或者简单解法。考虑到旋转角度参数通常为微小转角,还可以对旋转角参数作线性处理,从而将布尔莎、莫洛金斯基和武测模型简化为线性模型,大量的研究也主要围绕着线性模型展开^[9-12]。

综上所述,非线性模型存在矩阵严重病态、求解困难的问题,线性模型则只在小旋转角时适用,若旋转角稍大,极易导致解算的转换参数严重偏离真值。为了提高非线性模型的计算效率和计算精度,寻求一种简单易行的算法成为必需。鉴于粒子群算法作为高效的计算技术,已被证明可用于求解大量非线性、不可微和多峰值的复杂问题,而且其算法简洁,编制程序容易,已经被应用于多项科学和工程领域^[13]。特别是,相对于最小二乘法而言,粒子群算法不用求解矩阵,可以避免矩阵病态性带来的计算误差,同时可以避免最小二乘法求解过程中、非线性超定方程组线性化后带来的计算误差,再考虑到 Bursa 模型是应用主流,所以,本文拟采用粒子群算法建立 Bursa 非线性模型转换参数的求解方法,并通过实际算例证明其解的精度。

2 Bursa 非线性模型

设任意点 P 在坐标系 1 中的坐标为 X_1, Y_1, Z_1 , 在坐标系 2 中的坐标为 X_2, Y_2, Z_2 ; $\varepsilon_x, \varepsilon_y$ 和 ε_z 分别为 X_2, Y_2, Z_2 轴绕 X_1, Y_1, Z_1 轴的旋转角; $\Delta X, \Delta Y, \Delta Z$ 分别为两坐标系原点在 X, Y, Z 方向上的平移量; k 为尺度变化系数,则 Bursa 非线性模型可表达为:

$$\begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + (1+k) \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} \quad (1)$$

式中 $A_{11} = \cos \varepsilon_y \cos \varepsilon_z, A_{12} = \cos \varepsilon_x \sin \varepsilon_z + \sin \varepsilon_x \sin \varepsilon_y \cos \varepsilon_z, A_{13} = \sin \varepsilon_x \sin \varepsilon_z - \cos \varepsilon_x \sin \varepsilon_y \cos \varepsilon_z,$
 $A_{21} = -\cos \varepsilon_y \sin \varepsilon_z, A_{22} = \cos \varepsilon_x \cos \varepsilon_z - \sin \varepsilon_x \sin \varepsilon_y \sin \varepsilon_z,$
 $A_{23} = \sin \varepsilon_x \cos \varepsilon_z + \cos \varepsilon_x \sin \varepsilon_y \sin \varepsilon_z, A_{31} = \sin \varepsilon_y,$
 $A_{32} = -\sin \varepsilon_x \cos \varepsilon_y, A_{33} = \cos \varepsilon_x \cos \varepsilon_y。$

如果旋转角都为 0, 式(1)可简化为:

$$\begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + (1+k) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} \quad (2)$$

3 Bursa 模型的粒子群解法

3.1 建立 Bursa 模型的目标函数

设公共点 $i(i=1, \dots, n)$ 在坐标系 1 中的坐标为 X_{1i}, Y_{1i}, Z_{1i} , 根据公式(1), 推算其在坐标系 2 下的坐标 $X'_{2i}, Y'_{2i}, Z'_{2i}$:

$$\begin{cases} X'_{2i} = \Delta X + (1+k) (\cos \varepsilon_y \cos \varepsilon_z X_1 + \cos \varepsilon_x \sin \varepsilon_z Y_1 + \sin \varepsilon_x \sin \varepsilon_y \cos \varepsilon_z Y_1 + \sin \varepsilon_x \sin \varepsilon_z Z_1 - \cos \varepsilon_x \sin \varepsilon_y \cos \varepsilon_z Z_1) \\ Y'_{2i} = \Delta Y + (1+k) (-\cos \varepsilon_y \sin \varepsilon_z X_1 + \cos \varepsilon_x \cos \varepsilon_z Y_1 - \sin \varepsilon_x \sin \varepsilon_y \sin \varepsilon_z Y_1 + \sin \varepsilon_x \cos \varepsilon_z Z_1 + \cos \varepsilon_x \sin \varepsilon_y \sin \varepsilon_z Z_1) \\ Z'_{2i} = \Delta Z + (1+k) (\sin \varepsilon_y X_1 - \sin \varepsilon_x \cos \varepsilon_y Y_1 + \cos \varepsilon_x \cos \varepsilon_y Z_1) \end{cases} \quad (3)$$

设点 i 坐标系 2 下的真实坐标为 X_{2i}, Y_{2i}, Z_{2i} , 则可将粒子飞行的目标函数设置为:

$$\Delta f = \sum_{i=1}^n [(X'_{2i} - X_{2i})^2 + (Y'_{2i} - Y_{2i})^2 + (Z'_{2i} - Z_{2i})^2] \quad (4)$$

利用粒子群搜寻 7 参数, 带入公式(4), 当 $\Delta f \rightarrow 0$ 时, 所求 7 参数为最优解。

3.2 定义粒子的初始位置和初始速度

粒子的初始位置与速度具有随机性。设粒子有 m 个, $\varepsilon_x, \varepsilon_y, \varepsilon_z, \Delta X, \Delta Y, \Delta Z, k$ 的取值范围分别为 $[\varepsilon_{x1}, \varepsilon_{x2}], [\varepsilon_{y1}, \varepsilon_{y2}], [\varepsilon_{z1}, \varepsilon_{z2}], [\Delta X_1, \Delta X_2], [\Delta Y_1, \Delta Y_2], [\Delta Z_1, \Delta Z_2], [k_1, k_2]$ 。

第 j 个粒子的初始位置用二维数组表示:

$$\begin{cases} \varepsilon_x[j, 0] = \text{rand}() \times (\varepsilon_{x2} - \varepsilon_{x1}) + \varepsilon_{x1} \\ \varepsilon_y[j, 0] = \text{rand}() \times (\varepsilon_{y2} - \varepsilon_{y1}) + \varepsilon_{y1} \\ \varepsilon_z[j, 0] = \text{rand}() \times (\varepsilon_{z2} - \varepsilon_{z1}) + \varepsilon_{z1} \\ \Delta X[j, 0] = \text{rand}() \times (\Delta X_2 - \Delta X_1) + \Delta X_1 \\ \Delta Y[j, 0] = \text{rand}() \times (\Delta Y_2 - \Delta Y_1) + \Delta Y_1 \\ \Delta Z[j, 0] = \text{rand}() \times (\Delta Z_2 - \Delta Z_1) + \Delta Z_1 \\ k[j, 0] = \text{rand}() \times (k_2 - k_1) + k_1 \end{cases} \quad (5)$$

其中, $rand()$ 代表在范围 $[0,1]$ 内取值的随机函数; $\varepsilon_x[j,0]$ 、 $\varepsilon_y[j,0]$ 、 $\varepsilon_z[j,0]$ 、 $\Delta X[j,0]$ 、 $\Delta Y[j,0]$ 、 $\Delta Z[j,0]$ 、 $k[j,0]$ 分别代表第 j 个粒子不同参数下的初始位置,用二维数组表示, $j=1, \dots, m, 0$ 表示初始的意思,也是数组的第一列。

第 j 个粒子的初始速度用二维数组表示:

$$\begin{cases} V_{\varepsilon x}[j,0] = rand() \times (\varepsilon_{x2} - \varepsilon_{x1})/2 \\ V_{\varepsilon y}[j,0] = rand() \times (\varepsilon_{y2} - \varepsilon_{y1})/2 \\ V_{\varepsilon z}[j,0] = rand() \times (\varepsilon_{z2} - \varepsilon_{z1})/2 \\ V_{\Delta X}[j,0] = rand() \times (\Delta X_2 - \Delta X_1)/2 \\ V_{\Delta Y}[j,0] = rand() \times (\Delta Y_2 - \Delta Y_1)/2 \\ V_{\Delta Z}[j,0] = rand() \times (\Delta Z_2 - \Delta Z_1)/2 \\ V_k[j,0] = rand() \times (k_2 - k_1)/2 \end{cases} \quad (6)$$

其中, $rand()$ 代表在范围 $[0,1]$ 内取值的随机函数; $V_{\varepsilon x}[j,0]$ 、 $V_{\varepsilon y}[j,0]$ 、 $V_{\varepsilon z}[j,0]$ 、 $V_{\Delta X}[j,0]$ 、 $V_{\Delta Y}[j,0]$ 、 $V_{\Delta Z}[j,0]$ 、 $V_k[j,0]$ 分别代表第 j 个粒子不同参数下的初始速度,用二维数组表示, $j=1, \dots, m, 0$ 表示初始,也是数组的第一列。

3.3 飞行过程中更新粒子的位置和速度

粒子初始化以后,即开始寻找最优解。粒子每飞行一次,首先应该更新粒子位置。第一次飞行后,第 j 个粒子到达的新位置按“初始位置 + 1 次飞行时间 \times 飞行速度”计算,如:

$$\begin{cases} \varepsilon_x[j,1] = \varepsilon_x[j,0] + V_{\varepsilon x}[j,0] \\ \varepsilon_y[j,1] = \varepsilon_y[j,0] + V_{\varepsilon y}[j,0] \\ \varepsilon_z[j,1] = \varepsilon_z[j,0] + V_{\varepsilon z}[j,0] \\ \Delta X[j,1] = \Delta X[j,0] + V_{\Delta X}[j,0] \\ \Delta Y[j,1] = \Delta Y[j,0] + V_{\Delta Y}[j,0] \\ \Delta Z[j,1] = \Delta Z[j,0] + V_{\Delta Z}[j,0] \\ k[j,1] = k[j,0] + V_k[j,0] \end{cases} \quad (7)$$

其中, $\varepsilon_x[j,1]$ 、 $\varepsilon_y[j,1]$ 、 $\varepsilon_z[j,1]$ 、 $\Delta X[j,1]$ 、 $\Delta Y[j,1]$ 、 $\Delta Z[j,1]$ 、 $k[j,1]$ 分别代表第 j 个粒子不同参数下第一次飞行后到达的位置, $j=1, \dots, m, 1$ 表示飞行次数,也代表二维数组的第二列。为了保证粒子不飞出边界,还应该对其达到的位置做以下条件约束(以 ε_x 为例,其余类似):

$$\text{if } \varepsilon_x[j,1] < \varepsilon_{x1} \text{ or } \varepsilon_x[j,1] > \varepsilon_{x2}, \varepsilon_x[j,1] = \varepsilon_x[j,0] \quad (8)$$

其次更新粒子速度,粒子速度更新与粒子到达的位置距离目标的远近有关。依据公式(4),将粒子的初始参数 $\varepsilon_x[j,0]$ 、 $\varepsilon_y[j,0]$ 、 $\varepsilon_z[j,0]$ 、 $\Delta X[j,0]$ 、 $\Delta Y[j,0]$ 、 $\Delta Z[j,0]$ 、 $k[j,0]$ 和第一次飞行结束后的参数 $\varepsilon_x[j,1]$ 、 $\varepsilon_y[j,1]$ 、 $\varepsilon_z[j,1]$ 、 $\Delta X[j,1]$ 、 $\Delta Y[j,1]$ 、 $\Delta Z[j,1]$ 、 $k[j,1]$ 进行比较, Δf 较小的一组参数为粒

子 j 第一次飞行结束后搜索到的最好位置,用一维数组表示,并记为 $P_{\varepsilon x}[j]$ 、 $P_{\varepsilon y}[j]$ 、 $P_{\varepsilon z}[j]$ 、 $P_{\Delta X}[j]$ 、 $P_{\Delta Y}[j]$ 、 $P_{\Delta Z}[j]$ 、 $P_k[j]$;同时比较所有粒子的位置,并将所有粒子迄今为止搜索到的最好一组位置记为 $Q_{\varepsilon x}$ 、 $Q_{\varepsilon y}$ 、 $Q_{\varepsilon z}$ 、 $Q_{\Delta X}$ 、 $Q_{\Delta Y}$ 、 $Q_{\Delta Z}$ 、 Q_k ,则第 j 个粒子的速度在第一次飞行结束后变更为:

$$\begin{cases} V_{\varepsilon x}[j,1] = w \times V_{\varepsilon x}[j,0] + \\ \quad c_1 rand() \times (P_{\varepsilon x}[j] - \varepsilon_x[j,0]) + \\ \quad c_2 rand() \times (Q_{\varepsilon x} - \varepsilon_x[j,0]) \\ V_{\varepsilon y}[j,1] = w \times V_{\varepsilon y}[j,0] + \\ \quad c_1 rand() \times (P_{\varepsilon y}[j] - \varepsilon_y[j,0]) + \\ \quad c_2 rand() \times (Q_{\varepsilon y} - \varepsilon_y[j,0]) \\ V_{\varepsilon z}[j,1] = w \times V_{\varepsilon z}[j,0] + \\ \quad c_1 rand() \times (P_{\varepsilon z}[j] - \varepsilon_z[j,0]) + \\ \quad c_2 rand() \times (Q_{\varepsilon z} - \varepsilon_z[j,0]) \\ V_{\Delta X}[j,1] = w \times V_{\Delta X}[j,0] + \\ \quad c_1 rand() \times (P_{\Delta X}[j] - \Delta X[j,0]) + \\ \quad c_2 rand() \times (Q_{\Delta X} - \Delta X[j,0]) \\ V_{\Delta Y}[j,1] = w \times V_{\Delta Y}[j,0] + \\ \quad c_1 rand() \times (P_{\Delta Y}[j] - \Delta Y[j,0]) + \\ \quad c_2 rand() \times (Q_{\Delta Y} - \Delta Y[j,0]) \\ V_{\Delta Z}[j,1] = w \times V_{\Delta Z}[j,0] + \\ \quad c_1 rand() \times (P_{\Delta Z}[j] - \Delta Z[j,0]) + \\ \quad c_2 rand() \times (Q_{\Delta Z} - \Delta Z[j,0]) \\ V_k[j,1] = w \times V_k[j,0] + \\ \quad c_1 rand() \times (P_k[j] - k[j,0]) + \\ \quad c_2 rand() \times (Q_k - k[j,0]) \end{cases} \quad (9)$$

其中二维数组 $V_{\varepsilon x}[j,1]$ 、 $V_{\varepsilon y}[j,1]$ 、 $V_{\varepsilon z}[j,1]$ 、 $V_{\Delta X}[j,1]$ 、 $V_{\Delta Y}[j,1]$ 、 $V_{\Delta Z}[j,1]$ 、 $V_k[j,1]$ 分别代表第 j 个粒子不同参数下的速度,1 表示飞行次数为 1,也是数组的第二列; w 表示惯性权重,常数,用来控制粒子的历史速度对当前速度的影响程度,选取适当的值能平衡粒子群算法的全局和局部搜索能力,从而得到更好的解,本文取为 0.8; c_1 、 c_2 表示学习常数, c_1 为“认知”部分,表示粒子吸取自身经验知识的过程, c_2 为“社会”部分,表示粒子学习其他粒子经验的过程,表现了粒子间信息的共享与社会协作, c_1 、 c_2 本文都取 2。另外,为了避免粒子的飞行速度过快或过慢,粒子速度变化还应满足以下条件(以 ε_x 和 k 参数为例,其余类似):

$if\ V_{\varepsilon_x}[j,1]>(\varepsilon_{x2}-\varepsilon_{x1})/2,$

$V_{\varepsilon_x}[j,1]=(\varepsilon_{x2}-\varepsilon_{x1})/2$

$if\ V_{\varepsilon_x}[j,1]<-(\varepsilon_{x2}-\varepsilon_{x1})/2,$

$V_{\varepsilon_x}[j,1]=-(\varepsilon_{x2}-\varepsilon_{x1})/2$

...

$if\ V_k[j,1]>(k_2-k_1)/2,$

$V_k[j,1]=(k_2-k_1)/2$

$if\ V_k[j,1]<-(k_2-k_1)/2,$

$V_k[j,1]=-(k_2-k_1)/2$

(10)

由此,在飞行 t 次以后,第 j 个粒子的位置和速度分别变更为(以 ε_x 参数变更为例,其余类似):

$\varepsilon_x[j,t]=\varepsilon_x[j,t-1]+V_{\varepsilon_x}[j,t-1]$

$if\ \varepsilon_x[j,t]<\varepsilon_{x1}\ or$

$\varepsilon_x[j,t]>\varepsilon_{x2},\varepsilon_x[j,t]=\varepsilon_x[j,t-1]$

$V_{\varepsilon_x}[j,t]=w\times V_{\varepsilon_x}[j,t-1]+$

$c_1rand() \times (P_{\varepsilon_x}[j]-\varepsilon_x[j,t-1])+$

$c_2rand() \times (Q_{\varepsilon_x}-\varepsilon_x[j,t-1])$

$if\ V_{\varepsilon_x}[j,t]>(\varepsilon_{x2}-\varepsilon_{x1})/2,$

$V_{\varepsilon_x}[j,t]=(\varepsilon_{x2}-\varepsilon_{x1})/2$

$if\ V_{\varepsilon_x}[j,t]<-(\varepsilon_{x2}-\varepsilon_{x1})/2,$

$V_{\varepsilon_x}[j,t]=-(\varepsilon_{x2}-\varepsilon_{x1})/2$

(11)

需要注意的是,粒子 j 迄今为止搜索到的最好一组位置和所有粒子迄今为止搜索到的最好一组位置应该根据每次飞行的结果,按照公式(4)进行更新,如此才能更接近最优解的那组位置。

3.4 目标计算结果

如上所述,不断执行粒子的飞行过程,寻找当前群体内最优解,若优于历史最优解,则更新,当 $\Delta f \rightarrow 0$,则寻最优解过程结束,计算获得的 7 参数如下:

$\varepsilon_x=Q_{\varepsilon_x},\varepsilon_y=Q_{\varepsilon_y},\varepsilon_z=Q_{\varepsilon_z},\Delta X=Q_{\Delta X},\Delta Y=Q_{\Delta Y},\Delta Z=Q_{\Delta Z},k=Q_k$

(12)

4 Bursa 模型 7 参数实例解法

文献[6]采用的三维直角坐标转换最小二乘稳健估计法具有代表性,为了便于比较,以文献[6]中的实例作为本文的实例(表 1)。

采用的 $\varepsilon_x,\varepsilon_y,\varepsilon_z,\Delta X,\Delta Y,\Delta Z,k$ 的取值范围分别为 $[0,180^\circ],[0,180^\circ],[0,180^\circ],[-10\ 000,10\ 000],[-10\ 000,10\ 000],[-10\ 000,10\ 000],[-1,1],w=0.8,c_1=2,c_2=2$ 。采取不同方案的计算结果如表 2 所示。当粒子数为 1 000、飞行次数 10 000 时,计算误差为 1.89×10^{-12} ,显示出粒子群算法求解 7 参数具有极高的精度;2)不同方案下的参数计算结果存在较小的偏差,但是表中数据显示小的参数偏差会造成坐标转换误差的成倍增大;3)由于算法中存在随机函数,同一方案下重复计算会出现不同的计算结果,但是总的趋势不会发生改变,即计算精度随着粒子数和飞行次数的增大而显著提高。

按照方案 3,计算坐标与实际坐标的误差如表 3 所示。最大绝对误差为 0.000 4 mm,小于文献[6]中的最大绝对误差 20.001 mm,也小于文献[6]中 5、6、7、8 号点的最大绝对误差 0.001 mm。

5 结论

1)本文提出的 Bursa 模型 7 参数粒子群求解方法,其求解过程不需要解方程组,可以避免复杂的矩阵逆运算和方程矩阵病态问题,编程容易,且适合任何大角度旋转的坐标转换情况。

2)实例对比分析显示粒子群求解方法精度达到了 1.89×10^{-12} ,优于最小二乘稳健估计法计算结果,且随着粒子数量和飞行次数的增多,计算精度还会显著增大。

3)目前主流导航软件和 GPS 手持机均采用线性坐标转换,会带来坐标转换误差,而非线性坐标转换几乎可以消除算法本身带来的坐标转换误差。

表 1 模拟数据(单位:m)
Tab.1 Simulation data(unit:m)

点名	坐标系 1 下的坐标			坐标系 2 下的坐标		
	X_1	Y_1	Z_1	X_2	Y_2	Z_2
1	2 000	2 000	2 000	3 814.313 14	2 589.568 268	4 931.851 653
2	-2 000	2 000	2 000	2 400.099 577	5 039.058 011	2 103.424 528
3	-2 000	-2 000	2 000	-1 307.007 204	4 531.752 075	3 517.638 09
4	2 000	-2 000	2 000	107.206 359	2 082.262 332	6 346.065 215
5	2 000	2 000	-2 000	3 307.007 204	-531.752 075	2 482.361 91
6	-2 000	2 000	-2 000	1 892.793 641	1 917.737 668	-346.065 215
7	-2 000	-2 000	-2 000	-1 814.313 14	1 410.431 732	1 068.148 347
8	2 000	-2 000	-2 000	-400.099 577	-1 039.058 011	3 896.575 472

表 2 转换参数的计算结果

转换参数	方案 1		方案 2		方案 3	
	粒子数	飞行次数	粒子数	飞行次数	粒子数	飞行次数
	100	1 000	100	10 000	1 000	10 000
$\Delta X(\text{m})$	913.104 109 705 9		1 000.459 097 797 8		1 000.00	
$\Delta Y(\text{m})$	2 000.009 394 881 7		2 000.000 003 836 7		2 000.00	
$\Delta Z(\text{m})$	3 020.577 335 506 1		2 999.866 617 578 6		3 000.00	
$\varepsilon_x(^{\circ})$	26.979 644 731 4		29.846 934 797 1		29.999 999 997 0	
$\varepsilon_y(^{\circ})$	45.040 220 672 1		45.000 295 325 4		44.999 999 997 2	
$\varepsilon_z(^{\circ})$	62.148 704 156 4		60.117 058 627 8		60.000 000 000 5	
k	-4.59×10^{-4}		-1.20×10^{-6}		-1.00×10^{-11}	
$\Delta f(\text{m})$	152 649.697 151 627 0		231.726 795 823 4		1.89×10^{-12}	

表 3 计算坐标与实际值差异

点名	粒子群算法计算坐标(m)			坐标差(mm)		
	$X_{\text{算}}$	$Y_{\text{算}}$	$Z_{\text{算}}$	$X_{\text{算}} - X_2$	$Y_{\text{算}} - Y_2$	$Z_{\text{算}} - Z_2$
1	3 814.313 139 8	2 589.568 268 4	4 931.851 652 6	-0.000 19	0.000 43	-0.000 38
2	2 400.099 577 4	5 039.058 011 3	2 103.424 528 0	0.000 40	0.000 32	0.000 04
3	-1 307.007 203 7	4 531.752 075 0	3 517.638 090 3	0.000 26	-0.000 04	0.000 34
4	107.206 358 7	2 082.262 332 1	6 346.065 214 9	-0.000 34	0.000 07	-0.000 08
5	3 307.007 203 7	-531.752 075 0	2 482.361 909 7	-0.000 26	0.000 04	-0.000 34
6	1 892.793 641 3	1 917.737 667 9	-346.065 214 9	0.000 34	-0.000 07	0.000 08
7	-1 814.313 139 8	1 410.431 731 6	1 068.148 347 4	0.000 19	-0.000 43	0.000 38
8	-400.099 577 4	-1 039.058 011 3	3 896.575 472 0	-0.000 40	-0.000 32	-0.000 04

4)作为一种求解 7 参数的算法,和其他算法一样,其计算精度除了取决于算法本身的计算误差,更重要的还取决于公共点空间坐标质量的好坏。特别是当公共点较多时,若按各组点求解出的坐标参数偏差较大,则任何算法都将无法给出满足全部公共点坐标转换误差要求的满意结果。

5)粒子群算法规则简单,可调参数少,容易实现,特别是一些改进后的粒子群算法,收敛速度更快,且可以使算法避免陷入局部最优,对不同粒子群算法在 Bursa 模型 7 参数求解中的一个比较研究,也将作为本文后期的研究工作。

参 考 文 献

1 曾文宪,陶本藻. 三维坐标转换的非线性模型[J]. 武汉大学学报(信息科学版),2003,28(5):566-568. (Zheng Wenxian and Tao Benzao. Non-linear adjustment model of three-dimensional coordinate transformation[J]. Geomatics and Information Science of Wuhan University,2003,28(5):566-568)

2 罗长林,等. 基于改进的高斯-牛顿法的非线性三维直角坐标转换方法研究[J]. 大地测量与地球动力学,2007,(1):50-54. (Luo Changlin,et al. On nonlinear 3D rectan-

gular coordinate transformation method based on improved Gauss-Newton methord[J]. Journal of Geodesy and Geodynamics,2007,(1):50-54)

3 陈宇,白征东. 基于非线性最小二乘算法的空间坐标转换[J]. 大地测量与地球动力学,2010,(2):129-132. (Chen Yu and Bai Zhengdong. An nonlinear least squares algorithm for spatial coordinate transformation[J]. Journal of Geodesy and Geodynamics,2010,(2):129-132)

4 沈云中,胡雷鸣,李博峰. Bursa 模型用于局部区域坐标变换的病态问题及其解法[J]. 测绘学报,2006,35(2):95-98. (Shen Yunzhong, Hu Leiming and Li Bofeng. Ill-posed problem in determination of coordinate transformation parameters with small area's data based on Bursa model[J]. Acta Geodaetica et Cartographica Sinica, 2006,35(2):95-98)

5 陈义,沈云中. 非线性三维基准转换的稳健估计[J]. 大地测量与地球动力学,2003,(4):49-53. (Chen Yi and Shen Yunzhong. Robust estimation of three dimensional non-linear datum transformation[J]. Journal of Geodesy and Geodynamics,2003,(4):49-53)

6 罗长林,等. 三维直角坐标转换的一种阻尼最小二乘稳健估计法[J]. 武汉大学学报(信息科学版),2007,32(8): (下转第 94 页)