



(12) 发明专利

(10) 授权公告号 CN 103200670 B

(45) 授权公告日 2015.08.05

(21) 申请号 201310057842.6

(22) 申请日 2013.02.25

(73) 专利权人 北京科技大学

地址 100083 北京市海淀区学院路 30 号

(72) 发明人 杜利平 雷雨 康璐璐 姜少坤

(74) 专利代理机构 北京金智普华知识产权代理有限公司 11401

代理人 皋吉甫

(51) Int. Cl.

H04W 64/00(2009.01)

(56) 对比文件

CN 102752849 A, 2012.10.24,

US 8155673 B2, 2012.04.10,

于浩等. 加权凸集投影法在分布式目标定位

中的应用.《计算机工程》.2011, 全文.

审查员 孙珍珍

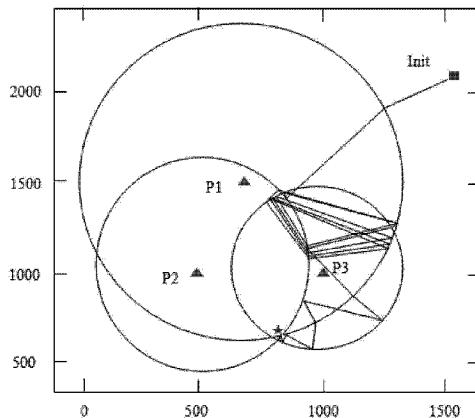
权利要求书2页 说明书6页 附图3页

(54) 发明名称

基于回溯检查凸集投影的认知无线电主用户定位方法

(57) 摘要

本发明提出一种基于回溯检查凸集投影的认知无线电主用户定位方法,以 L 个感知用户的坐标为圆心,利用凸集投影算法对主用户进行 Mc 步正交投影迭代;进行 mc 步回溯比较检查,计算出相邻迭代点之间的距离,并与门限值  $\lambda$  进行比较;若相邻迭代点之间的距离存在部分为零或大于  $\lambda$  的情况,则进行凸集圆域边界上 Mb 步投影迭代,和 mb 步回溯比较检查,计算相邻两个迭代点间的距离值,再次与门限值  $\lambda$  进行比较,若均小于  $\lambda$ ,则将 Mb 步的迭代结果  $b_{Mb}$  确定为主用户位置信息的定位结果。本发明中的回溯检查凸集投影定位算法弥补了现有凸集投影定位算法的不足,定位算法好,而且受测距误差的影响较小,适用于认知无线电网络中感知用户对主用户位置信息的获取环节。



1. 一种认知无线电网络中对主用户定位的方法,其特征在于,该方法包括以下步骤:

步骤一、以 L 个感知用户的坐标为圆心,利用凸集投影算法对主用户进行 Mc 步正交投影迭代,得到 Mc 个迭代点  $x_k$ ,其中  $k = 1, 2, 3, \dots, Mc$ ;

步骤二. 对步骤一中获取的 Mc 个迭代点,进行  $mc$  步回溯比较检查,计算出相邻迭代点之间的距离  $||x_{m+1}-x_m||$ ,其中,  $m = Mc-1, \dots, Mc-mc$ ,并与门限值  $\lambda$  进行比较;

步骤三. 如果步骤二中的回溯比较检查中,相邻迭代点之间的距离均小于  $\lambda$  且不为零,则将步骤一中最后 L 个迭代均值作为主用户位置信息的定位结果;如果步骤二中的回溯比较检查中,相邻迭代点之间的距离存在部分为零或大于  $\lambda$  的情况,继续执行步骤四;

步骤四. 以  $Mc$  步迭代结果  $x_{Mc}$  为初始点  $b_0$ ,进行凸集圆域边界上正交投影迭代,迭代检查步数为  $Mb$ ,得到  $Mb$  个迭代点  $b_h$ ,其中  $h = 1, 2, 3, \dots, Mb$ ;

步骤五. 对步骤四中获取的  $Mb$  个迭代点,进行  $mb$  步回溯比较检查,计算相邻两个迭代点间的距离值  $||b_{n+1}-b_n||$ ,其中,  $n = Mb-1, \dots, Mb-mb$ ,并与门限值  $\lambda$  进行比较;

步骤六. 如果步骤五中的回溯比较检查中,相邻迭代点间的距离值均小于  $\lambda$ ,则将步骤四中最后 L 个迭代均值作为主用户位置信息的定位结果;如果步骤五中的回溯比较检查中,相邻迭代点间的距离值存在大于  $\lambda$  的情况,跳转到步骤四以  $Mb$  步迭代结果  $b_{Mb}$  为初始点  $b_0$ ,并变换投影迭代顺序,直到相邻迭代点间的距离值均小于  $\lambda$ ;

所述步骤一包括:

1. 1) 初始化步骤:设置初始点  $x_0$ ,其中  $x_0$  为任意位置上的一点;

1. 2) 利用以下公式进行投影迭代:

$$x_{k+1} = P_{D_i}(x_k) = \begin{cases} x_k & x_k \subseteq D_i \\ P_i + d_i[\cos(\varphi), \sin(\varphi)]^T & x_k \not\subseteq D_i \end{cases}$$

,  $k = 0, 1, 2, \dots, Mc-1$

其中,  $P_{D_i}(x_k)$  表示正交凸集投影点,  $d_i[\cos(\varphi), \sin(\varphi)]^T$  表示  $P_i$  到  $P_{i+1}$  的向量; $P_i$  为第 i 个感知用户的位置坐标,  $i \in [1, L]$ ;  $D_i$  为以第 i 个感知用户为圆心,以第 i 个感知用户测得的与主用户之间的距离测量值为半径的凸集圆域。

2. 如权利要求 1 所述的认知无线电网络中对主用户定位的方法,其特征在于,  $\lambda$  的取值大小取决于感知用户对主用户测距结果的平均值,且  $\lambda$  相对于该平均距离是一个很小值。

3. 如权利要求 1 所述的认知无线电网络中对主用户定位的方法,其特征在于, $mb$  与  $mc$  取值相同,为 L 的整数倍。

4. 如权利要求 1 所述的认知无线电网络中对主用户定位的方法,其特征在于,所述步骤四包括:

4. 1) 初始化步骤:1) 设置初始点  $b_0$ ,  $b_0 = x_{Mc}$

4. 2) 利用以下公式进行投影迭代:

$$b_{h+1} = P_{h \bmod L}(b_h), h = 0, 1, 2, 3, \dots, Mb-1$$

其中,设  $h \bmod L$  为  $i$ ,  $b_h$  为  $y$ ,则  $P_i(y) = \arg \min_{x \in G_i} \|y - x\| = P_i + d_i * \frac{y - P_i}{\|y - P_i\|}$

其中,  $P_i$  为第 i 个感知用户的位置坐标,  $i \in [1, L]$ ;  $d_i$  是第 i 个感知用户测得的与主

用户之间的距离测量值 ; $C_i = \{y \in R^2 : ||y - P_i|| = d_i\}$  是第  $i$  个感知用户所确定的半径为  $d_i$  的圆边界。

## 基于回溯检查凸集投影的认知无线电主用户定位方法

### 技术领域

[0001] 本发明涉及一种认知无线电网络中对主用户定位的方法，特别涉及一种基于回溯检查的凸集投影定位方法。

### 背景技术

[0002] 随着无线移动通信与计算机网络的结合应用发展越来越成熟，移动互联网已经称为当今世界发展速度最快、市场潜力最大，商业价值最高的发展业务之一。丰富的应用主要依托于文字、图像、视频等信息承载方式，而且随着人们对应用质量的不断追求，要求信息的传递越来越高效、便捷。这些应用的创新与发展需要较宽频谱和较高下载速率。认知无线电的概念迎合了频谱重复利用的需要，通过对无线环境的感知实现与主用户的冲突避免，利用最优化的决策有效的动态利用频谱空洞。如果能够获取主用户的位置信息，那么对频谱感知的性能将会得到很大提升，并且在后续的频谱资源的管理与分配中也将起到很大的帮助作用。

[0003] 在认知网络中对主用户进行定位，获取主用户的位置信息的主要作用有以下几个方面：

[0004] 1. 为频谱资源管理提供支持。在主用户位置信息已知的情况下，根据其位置信息可以更好地提高频谱利用率，更好地指导感知用户不干扰主用户的频谱使用。

[0005] 2. 减小认知网络中用户的功耗。在主用户位置信息已知的情况下，认知网络中的感知用户可以根据主用户的位置信息来确定频谱感知的方向，在最小功率的运行状态下，便可以准确判断主用户的频谱使用情况。

[0006] 3. 避免对主用户的干扰。在主用户位置信息已知的情况下，可以结合多天线技术，针对主用户的方向位置进行频谱感知，避免了频谱间相互干扰的可能性。

[0007] 4. 有利于感知用户的位置优化。在主用户位置信息已知的情况下，根据主用户的位置信息，可以合理的分布感知用户的位置，提高频谱和空间的利用率，更好地避免对主用户干扰。

[0008] 目前常用的凸集投影方法包括 Circular POCS, Hyperbolic POCS, Boundary POCS 和 Hybrid POCS 等，其中 Hybrid POCS 是前两种 POCS 方法的合并，根据研究结果表明，Hybrid POCS 方法的定位精度要优于前几种方法，然而，当主用户远离感知用户时，由于 Hybrid POCS 算法中双曲线投影定位对于主用户在感知用户多边形之外的情况下收敛点受噪声波动较大，因此误差随着测距增大而增大。

### 发明内容

[0009] 本发明旨在解决上述技术缺陷，提出一种应用于认知网络中对主用户进行定位的回溯凸集投影算法 (BackCheck POCS)。

[0010] 该方法包括以下步骤：

[0011] 步骤一、以 L 个感知用户的坐标为圆心，利用凸集投影算法对主用户进行 Mc 步正

交投影迭代,得到  $M_c$  个迭代点  $x_k$ ,其中  $k = 1, 2, 3, \dots, M_c$ ;

[0012] 步骤二. 对步骤一中获取的  $M_c$  个迭代点,进行  $m_c$  步回溯比较检查,计算出相邻迭代点之间的距离  $||x_{m+1} - x_m||$ ,其中,  $m=M_c-1, \dots, M_c-m_c$

[0013] 步骤三. 如果步骤二中的回溯比较检查中,相邻迭代点之间的距离均小于  $\lambda$  且不为零,则将步骤一中最后  $L$  个迭代均值作为主用户位置信息的定位结果;如果步骤二中的回溯比较检查中,相邻迭代点之间的距离存在部分为零或大于  $\lambda$  的情况,继续执行步骤四;

[0014] 步骤四. 以  $M_c$  步迭代结果  $x_{M_c}$  为初始点  $b_0$ ,进行凸集圆域边界上正交投影迭代,迭代检查步数为  $M_b$ ,得到  $M_b$  个迭代点  $b_h$ ,其中  $h=1, 2, 3, \dots, M_b$ ;

[0015] 步骤五. 对步骤四中获取的  $M_b$  个迭代点,进行  $m_b$  步回溯比较检查,计算相邻两个迭代点间的距离值  $||b_{n+1} - b_n||$ ,其中,  $n=M_b-1, \dots, M_b-m_b$ ,并与门限值  $\lambda$  进行比较;

[0016] 步骤六. 如果步骤五中的回溯比较检查中,相邻迭代点间的距离值均小于  $\lambda$ ,则将步骤四中最后  $L$  个迭代均值作为主用户位置信息的定位结果;如果步骤五中的回溯比较检查中,相邻迭代点间的距离值存在大于  $\lambda$  的情况,跳转到步骤四以  $M_b$  步迭代结果  $b_{M_b}$  为初始点  $b_0$ ,并变换投影迭代顺序,直到相邻迭代点间的距离值均小于  $\lambda$ 。

[0017] 优选地,所述步骤一包括:

[0018] 1. 1) 初始化步骤:设置初始点  $x_0$ ,其中  $x_0$  为任意位置上的一点;

[0019] 1. 2) 利用以下公式进行投影迭代:

[0020]

$$x_{k+1} = P_{D_i}(x_k) = \begin{cases} x_k & x_k \subseteq D_i \\ P_i + d_i [\cos(\varphi), \sin(\varphi)]^T & x_k \not\subseteq D_i \end{cases}$$

[0021]

$$, \quad k=0,1,2..M_c-1$$

[0022] 其中,  $P_{D_i}(x_k)$  表示正交凸集投影点,  $d_i [\cos(\varphi), \sin(\varphi)]^T$  表示  $P_i$  到  $P_{i+1}$  的向量;  $P_i$  为第  $i$  个感知用户的位置坐标,  $i \in [1, L]$ 。  $D_i$  为以第  $i$  个感知用户为圆心,以第  $i$  个感知用户测得的与主用户之间的距离测量值为半径的凸集圆域。

[0023] 优选地,  $\lambda$  的取值大小取决于感知用户对主用户测距结果的平均值,且  $\lambda$  相对于该平均距离是一个很小值。

[0024] 优选地,  $m_b$  与  $m_c$  取值相同,为  $L$  的整数倍。

[0025] 优选地,所述步骤四包括:

[0026] 4. 1) 初始化步骤:1) 设置初始点  $b_0$ ,  $b_0 = x_{M_c}$

[0027] 4. 2) 利用以下公式进行投影迭代:

[0028]  $b_{h+1} = P_{hmodL}(b_h)$ ,  $h = 0, 1, 2, 3 \dots M_b-1$

[0029] 其中,  $P_i(y) = \arg \min_{x \in Q} \|y - x\| = P_i + d_i * \frac{y - P_i}{\|y - P_i\|}$

[0030] 其中,  $P_i$  为第  $i$  个感知用户的位置坐标,  $i \in [1, L]$ 。  $d_i$  是第  $i$  个感知用户测得的与主用户之间的距离测量值;  $C_i = \{y \in \mathbb{R}^2 : \|y - P_i\| = d_i\}$  是第  $i$  个感知用户所确定的半径为  $d_i$  的圆边界。

[0031] 该算法基于凸集投影定位算法的改进,弥补了现有凸集投影定位算法的不足,受

测距误差的影响相对较小,适合应用于认知无线电网络中感知用户对主用户位置信息的获取环节,能够更准确的实现对主用户的定位。

### 附图说明

- [0032] 图 1 为本发明中基于 BackCheck POCS 定位方法的流程图。
- [0033] 图 2 为本发明基于 BackCheck POCS 定位方法的迭代示意图。
- [0034] 图 3 为 BackCheck POCS 和 Hybrid POCS 的定位误差对比示意图。
- [0035] 图 4 为不同测距误差对 BackCheck POCS 及 Hybrid POCS 定位误差影响的对比示意图。
- [0036] 下面通过参考附图描述的实施例是示例性的,仅用于解释本发明,而不能解释为对本发明的限制。

[0037] 假设有 L 个感知用户参与对主用户的定位,L 个感知用户的位置坐标是已知的,表示为 :

- [0038]  $\{P_i = (X_i, Y_i) \in \mathbb{R}^2\}_{i=1}^L$
- [0039] L 个感知用户测得的与主用户之间的距离测量值表示为 :
- [0040]  $\{d_i > 0\}_{i=1}^L$
- [0041] 以每个感知用户为圆心,以距离测量值  $d_i$  为半径的凸集圆域表示为 :
- [0042]  $\{D_i \subset \mathbb{R}^2\}_{i=1}^L$
- [0043] 第 i 个感知用户所确定的半径为  $d_i$  的圆边界表示如下 :
- [0044]  $C_i = \{y \in \mathbb{R}^2 : \|y - P_i\| = d_i\}$
- [0045] 本发明定位方法所用的 POCS 算法可以是 Circular POCS, Hyperbolic POCS 和 Boundary POCS。现以 Circular POCS 算法为例,给出基于 BackCheck POCS 算法的具体步骤 :

[0046] 步骤一. 以 L 个感知用户的坐标为圆心,利用 Circular POCS 对主用户进行  $M_c$  步正交投影迭代,得到  $M_c$  个迭代点  $x_1, x_2, \dots, x_{M_c-1}, x_{M_c}$ 。

[0047] 在此设定 Circular POCS 的迭代检查步数为  $M_c$ ,由于 Circular POCS 的收敛速度很快,  $M_c$  的取值可以适当取一个较小值。

[0048] 根据 Circular POCS 的投影迭代规则,对主用户进行  $M_c$  步正交投影迭代的步骤为 :

- [0049] 1) 初始化 : 设置初始点  $x_0$ ,其中  $x_0$  为平面内任意一点,如图 2 中方块所示 ;
- [0050] 2) 进行简化的 circular POCS 的正交投影迭代 :
- [0051]

$$x_{k+1} = P_{D_i}(x_k) = \begin{cases} x_k & x_k \subseteq D_i \\ P_i + d_i[\cos(\varphi), \sin(\varphi)]^T & x_k \not\subseteq D_i \end{cases}$$

[0052]

$$, \quad k = 0, 1, 2, \dots, M_c - 1$$

[0053] 其中  $P_{D_i}(x_k)$  表示正交凸集投影点,  $d_i[\cos(\varphi), \sin(\varphi)]^T$  表示  $P_i$  到  $P_{i+1}$  的向量,可以通过 i 表示出迭代的顺序,此顺序是根据感知用户的顺序确定 ;当后续迭代进入死循

环或者不收敛时,可改变迭代的顺序,继续迭代。

[0054] 步骤二. 对步骤一中获取的  $M_c$  个迭代点,进行  $m_c$  步回溯检查。计算相邻迭代点之间的距离  $||x_{m+1}-x_m||$ ,其中,  $m=M_c-1, \dots, M_c-m_c$ ,并与门限值  $\lambda$  进行比较。其中,  $\lambda$  的取值大小取决于感知用户对主用户测距结果的平均值,且  $\lambda$  相对于该平均距离是一个很小值,例如:  $\lambda$  与该平均距离的比值小于等于 0.02,考虑到算法的运算复杂度,可将该比值进一步限定在 0.005~0.02 内。

[0055] 回溯检查步数为  $m_c$ ,该  $m_c$  的取值为参与定位的感知用户的数目  $L$  的整数倍。

[0056] 步骤三. 如果步骤二中的回溯比较检查中,相邻迭代点之间的距离均小于  $\lambda$  且不为零,则可以判断主用户位于感知用户所形成的多边形之内,将步骤一中最后  $L$  个迭代均值,即感知用户所属各个凸集的最后一次迭代的均值  $\hat{P} = \frac{x_{M_c} + x_{M_c-1} + \dots + x_{M_c-L+1}}{L}$  作为主用户的位置信息的定位结果;如果步骤二中的回溯比较检查中,相邻迭代点之间的距离存在部分为零或大于  $\lambda$  的情况,继续执行步骤四。

[0057] 步骤四. 以第  $M_c$  步迭代结果  $x_{M_c}$  为初始点  $b_0$ ,进行凸集圆域边界上正交投影迭代,迭代检查步数为  $M_b$ ,得到  $M_b$  个迭代点。

[0058] 设定边界正交投影迭代的检查步数为  $M_b$ ,由于取消了被迭代点位置的判断,边界正交投影迭代的收敛速度是不确定的,可能会很快收敛到主用户附近,也可能陷入缓慢循环,因此  $M_b$  取较大值,使边界正交投影迭代充分。

[0059] 其中,根据边界正交投影迭代的规则,对主用户进行  $M_b$  步的迭代步骤为:

[0060] 1) 设置初始点  $b_0$ ,  $b_0 = x_{M_c}$

[0061] 2)  $b_{h+1} = P_{h \text{ mod } L}(b_h)$ ,  $h = 0, 1, 2, 3, \dots, M_b-1$

[0062] 其中,  $P_i(y) = \arg \min_{x \in Q} \|y - x\| = P_i + d_i * \frac{y - P_i}{\|y - P_i\|}$

[0063] 步骤五. 对步骤四中获取的  $M_b$  个迭代点,进行回溯  $m_b$  步检查。计算相邻两个迭代点间的距离值,并与门限值  $\lambda$  进行比较。

[0064] 回溯检查步数为  $m_b$ ,该  $m_b$  取值与  $m_c$  相同,即为  $L$  的整数倍。

[0065] 步骤六. 如果步骤五中的回溯比较检查中,相邻迭代点间的距离值均小于  $\lambda$ ,则将步骤四中最后  $L$  个迭代均值,即感知用户所属各个凸集的最后一次迭代的均值  $\hat{P} = \frac{b_{M_b} + b_{M_b-1} + \dots + b_{M_b-L+1}}{L}$  作为主用户的位置信息的定位结果;如果步骤五中的回溯比较

检查中,相邻迭代点间的距离值存在大于  $\lambda$  的情况,说明边界正交投影在经过了  $M_b$  步充分迭代之后,依然没有收敛到主用户位置附近,而是陷入了缓慢循环迭代。此时,跳转步骤四继续执行凸集圆域边界上的正交投影迭代,其中,以前次  $M_b$  步正交投影迭代中第  $M_b$  步迭代结果  $b_{M_b}$  为初始点  $b_0$ ,变换原有的投影迭代顺序,直到相邻迭代点间的距离值均小于  $\lambda$ 。

[0066] 以下结合附图和具体的实例来对本发明做进一步的详细说明。

[0067] 步骤一. 以  $L$  个感知用户的坐标为圆心,利用 Circular POCS 对主用户进行  $M_c$  步正交投影迭代,得到  $M_c$  个迭代点  $x_k$ 。

[0068] 设定感知用户数目为  $L=3$ ,感知用户的位置坐标为  $[(700m, 1500m), (500m, 1000m), (1000m, 1000m)]$ 。其中,输入白噪声作为感知用户获取的与主用户之间距离的测量值  $d_i$ ,

方差为 8m。设定圆凸集投影迭代步数上线  $M_c=10$ , 以坐标位置  $x_0=(1600m, 2100m)$  为起始位置(如图 2 方块所示) 进行投影迭代, 得到 10 个迭代点  $x_k, k=1, 2, 3, \dots, 10$ 。

[0069] 步骤二. 对步骤一中获取的 10 个迭代点, 进行  $m_c$  步回溯检查, 回溯检查步数为  $m_c=2L=6$ 。计算相邻迭代点之间的距离, 并与门限值  $\lambda$  进行比较。假设所有参与定位的感知用户所获取的与主用户之间距离的测量值的平均值为  $R$ , 则检查门限值  $\lambda$  设定为相对  $R$  的一个较小量, 这里设定  $\lambda$  与距离平均值的比值为  $\lambda/R=0.01$ 。

[0070]  $||x_{m+1}-x_m|| \leq \lambda$ , 其中,  $m=M_c-1, \dots, M_c-m_c$

[0071] 步骤三. 由于步骤二中的回溯比较检查中, 相邻迭代点之间的距离存在部分为零或大于  $\lambda$  的情况, 继续执行步骤四。

[0072] 从图 2 中可以看出, 从初始点  $x_0$  开始经过两步 Circular POCS 的迭代, 迭代点便停滞于三个凸集圆域的交集区域上, 此时根据回溯检查判断的结果, 迭代点的变化差值存在一部分为零的情况, 因而继续执行步骤四, 进行向凸集圆域边界上的正交投影迭代。

[0073] 步骤四. 以第  $M_c$  步迭代结果  $x_{M_c}$  为初始点, 进行凸集圆域边界上的正交投影迭代对主用户进行  $M_b$  步正交投影迭代, 得到  $M_b$  个迭代点  $b_h, h=1, 2, 3, \dots, M_b$ 。

[0074] 首先, 设定边界正交投影迭代步数上限  $M_b=30$ , 由于取消了被迭代点位置的判断, 边界正交投影迭代的收敛速度是不确定的, 可能会很快收敛到主用户附近, 也可能陷入缓慢循环, 因此  $M_b$  取较大值, 使边界正交投影迭代充分; 之后, 根据边界正交投影迭代的规则, 对主用户进行  $M_b$  步定位投影, 得到 10 个迭代点  $b_h, h=1, 2, 3, \dots, 30$ 。

[0075] 步骤五. 对步骤四中获取的  $M_b$  个迭代点, 进行回溯  $m_b$  步检查。回溯检查的步数  $m_b=2L=6$ 。计算相邻两个迭代点间的距离值, 并与门限值  $\lambda$  进行比较。

[0076] 步骤六. 在迭代了  $M_b$  步之后, 回溯检查  $m_b=6$  步的迭代点变化值, 发现相邻迭代点间的距离值存在大于  $\lambda$  的情况, 投影迭代陷入了缓慢循环投影迭代, 因此, 需要跳转到步骤四, 并以首次  $M_b$  步正交投影迭代中第  $M_b$  步的迭代结果为初始点  $b_0$ , 再次执行凸集圆域边界上的  $M_b$  步正交投影迭代, 此时变换首次的投影迭代顺序, 将原投影迭代的顺序由 P1-P2-P3 改为 P2-P3-P1, 当再次经过  $M_b$  步迭代, 回溯检查  $m_b=6$  步, 发现迭代点的变化值小于门限  $\lambda$ , 说明迭代收敛主用户位置附近, 因此将第二次  $M_b$  步正交投影迭代中最后  $L$  个迭代均值(如图 2 星号所示) 确定主用户位置。

[0077] 图 3 是 Hybrid POCS 定位算法与 BackCheck POCS 定位算法的定位仿真结果比较图。图中横坐标为仿真重复次数, 纵坐标是估计位置与目标真实位置之间的差值与感知用户到主用户之间真实距离平均值的比值。从图 3 中可以看出, 一般情况下两种算法的定位精度比较接近, 但是在某些情况下, BackCheck POCS 定位算法的定位精度比较有优势。这是因为, 当主用户远离感知用户时, 双曲线的渐近线性质容易致使双曲线的交点受到测距噪声的波动影响比较明显, 因此在这种情况下, 可以看出 BackCheck POCS 算法的定位效果要比 Hybrid POCS 算法优越。

[0078] 图 4 描述了 Hybrid POCS 定位算法与 BackCheck POCS 定位算法在不同测距误差影响下, 定位精度的比较。从图 4 中可以看出, BackCheck POCS 定位算法比 Hybrid POCS 定位算法具有一定的优势, 这主要是由于 Hybrid POCS 算法中双曲线投影定位对于主用户在感知用户多边形之外的情况下收敛点受噪声波动较大, 因此随着测距误差的增大。

[0079] 尽管已经示出和描述了本发明的实施例, 对于本领域的普通技术人员而言, 可以

理解在不脱离本发明的原理和精神的情况下可以对这些实施例进行多种变化、修改、替换和变型，本发明的范围由所附权利要求及其等同限定。

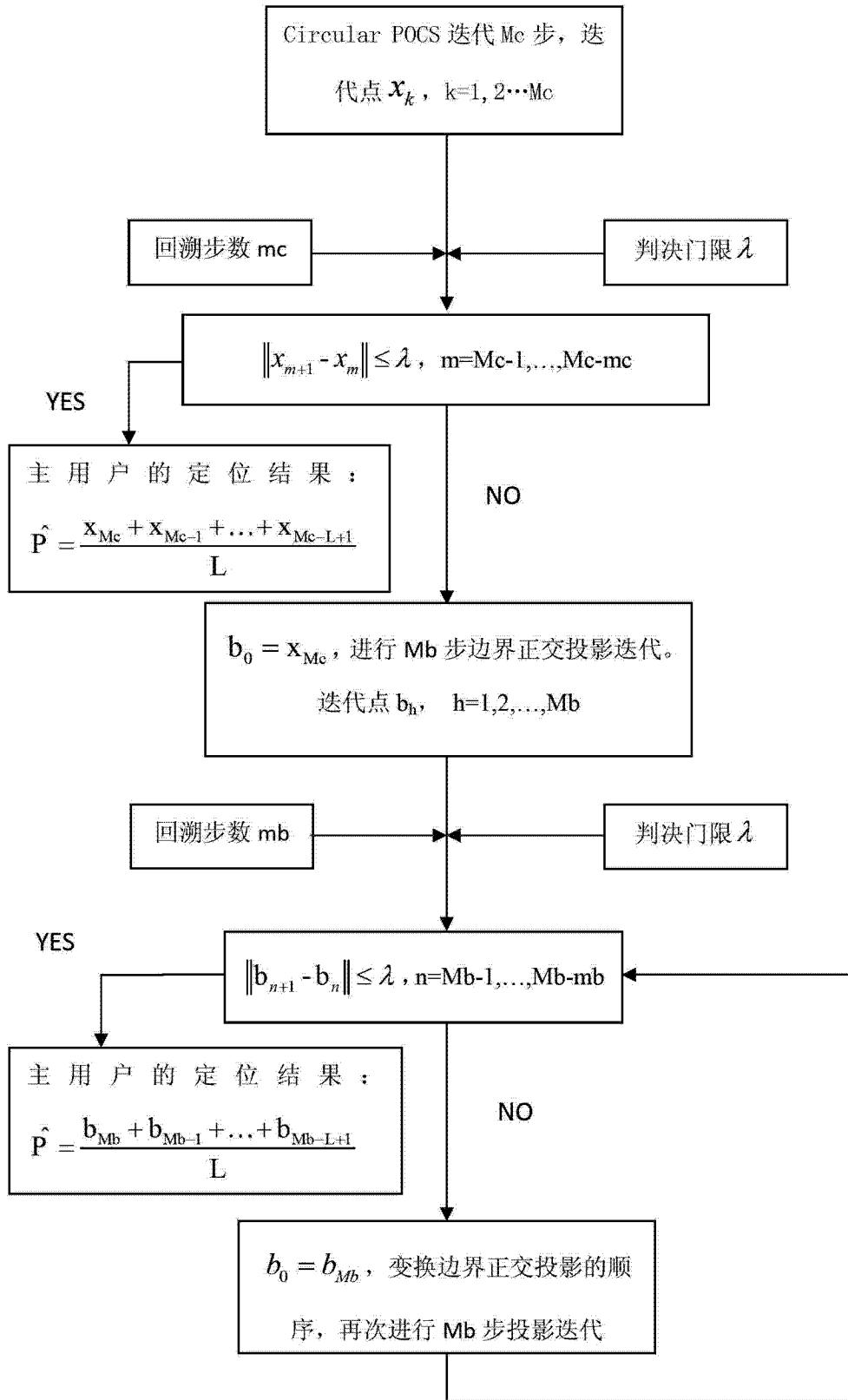


图 1

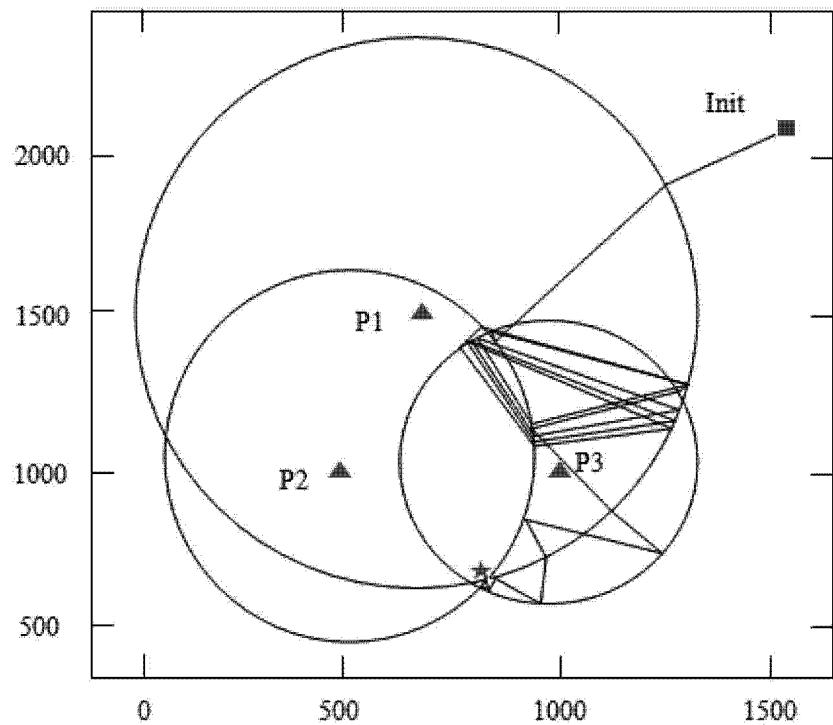


图 2

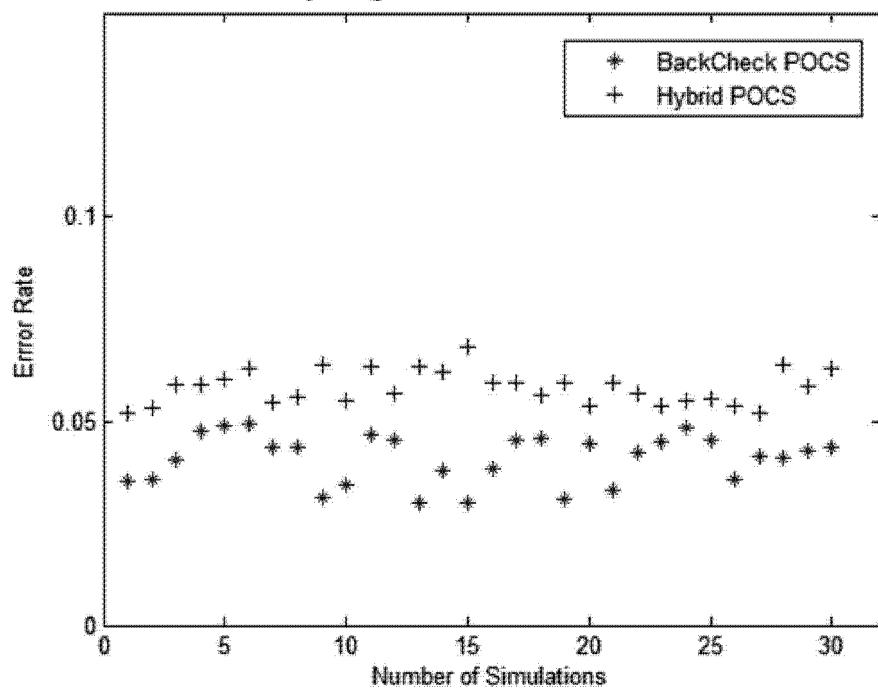


图 3

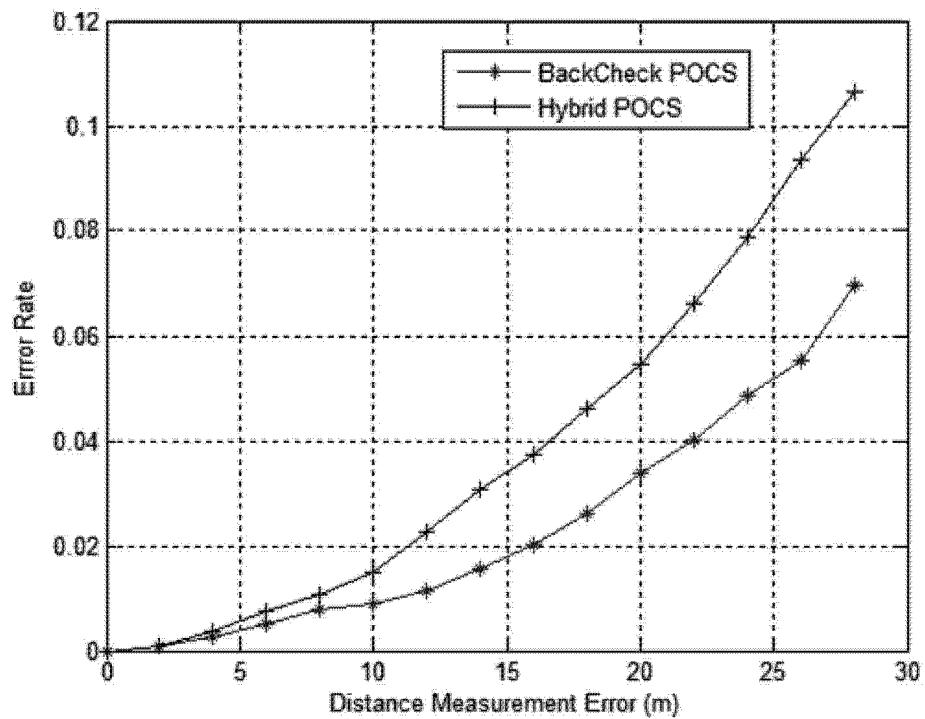


图 4