(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 113054948 A
(43)申请公布日 2021.06.29

(21)申请号 202011358316.X

(22)申请日 2020.11.27

(71)申请人 江苏亨通太赫兹技术有限公司
地址 215200 江苏省苏州市吴江经济技术开发区亨通路88号
申请人 北京亨邮太赫兹通信技术有限公司

(72)发明人 曹会扬 姚远 孙昌达 刘元智 侯树海

(74)专利代理机构 苏州市中南伟业知识产权代理事务所(普通合伙) 32257
代理人 殷海霞

(51)Int.Cl.
H03H 17/02(2006.01)

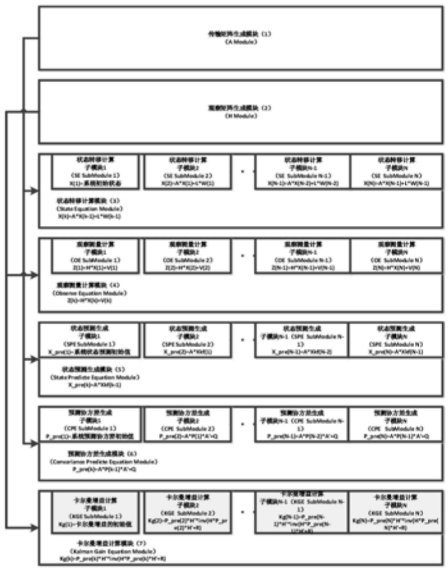
权利要求书8页 说明书31页 附图8页

(54)发明名称

并行卡尔曼滤波系统和方法

(57)摘要

本发明公开了一种并行卡尔曼滤波方法，包括以下步骤，将卡尔曼的运算流程拆分为多个步骤，在使每个步骤之间进行并行运算的同时，也使每个步骤用多个并行的子模块进行同时运算。并行卡尔曼滤波系统，包括传输矩阵生成模块、观察矩阵生成模块、状态转移计算模块、观察测量计算模块、状态预测生成模块、预测协方差生成模块、卡尔曼增益计算模块、状态最优估算值计算模块、状态协方差更新模块、测量过程中的噪声生成模块、状态转移过程中产生的噪声生成模块、过程噪声协方差生成模块和测量噪声协方差生成模块。其能够有效的提升卡尔曼滤波的运行效率，可以同时实现多个状态的卡尔曼滤波，也兼顾了计算效率与计算资源之间的平衡问题。



CN 113054948 A

1.一种并行卡尔曼滤波方法，其特征在于：包括以下步骤，将卡尔曼的运算流程拆分为多个步骤，在使每个步骤之间进行并行运算的同时，也使每个步骤用多个并行的子模块进行同时运算。

2.如权利要求1所述的并行卡尔曼滤波方法，其特征在于：卡尔曼运算流程至少拆分成以下步骤：S1，生成传输矩阵A；S2，生成观察矩阵H；S3，计算系统的状态转移；S4，计算观察测量值；S5，生成状态预测；S6，生成预测协方差P_pre(k)；S7，计算卡尔曼增益；S8，计算状态最优估算值；S9，更新状态协方差；S10，产生测量过程中遇到的噪声V和状态转移过程中产生的噪声W。

3.如权利要求2所述的并行卡尔曼滤波方法，其特征在于：包括以下步骤，

(A)计算传输矩阵A：取A为n*n维矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$；

(B)计算观察矩阵H：取H为n*n维矩阵 $H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$；

(C)计算系统的状态值：由状态转移方程X(k)＝A*X(k-1)+L*W(k-1)计算系统的状态转移；

(D)计算观察测量值：采用步骤(B)中计算出的矩阵 $H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$ 和第k个状态矩阵 $X(K)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_k$ 相乘再和第k个状态下的测量噪声矩阵V(K)相加得到第k个状态下的测量值Z(K)，观察矩阵的计算方程为：Z(k)＝H*X(k)+V(k)；

(E)生成状态预测：采用步骤(A)计算传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$ 和步骤(H)计算出的第k-1个状态的最优估计值Xkf(k-1)，做矩阵乘法得到第k个状态的预测值X_pre(k)，取Xkf(k-1)为n维列向量 $Xkf(k-1)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_{k-1}$，取X_pre(k)为n维列向量 $X_pre(k)=\begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_k$；状态预测的计算方程为：X_pre(k)＝A*Xkf(k-1)；

(F)生成预测协方差P_pre(k)：取P_pre(k)为n*n维矩阵 $P_pre(k)=\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k$，其计算方程为P_pre(k)＝A*P(k-1)*A'+Q；

(G)计算卡尔曼增益:取卡尔曼增益Kg为n维列向量 $Kg=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}$,其计算方程为Kg(k)=P_pre(k)*H'*inv(H*P_pre(k)*H'+R);

(H)计算状态最优估算值:取状态最优估算值 $Xkf(k)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_k$,其计算方程为Xkf(k)=X_pre(k)+Kg(k)*(Z(k)-H*X_pre(k));

(I)更新状态协方差:其计算方程为P(k)=(I-Kg(k)*H)*P_pre(k);

(J)(K)分别为是为了产生测量过程中遇到的噪声和状态转移过程中产生的噪声 $V(K)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_k$ 和 $W(K)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_k$,这两个噪声的协方差分别为R和 $Q=\begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix}$。

4.一种并行卡尔曼滤波系统,其特征在于:使用如权利要求1-3任意一项所述的并行卡尔曼滤波方法,包括传输矩阵生成模块、观察矩阵生成模块、状态转移计算模块、观察测量计算模块、状态预测生成模块、预测协方差生成模块、卡尔曼增益计算模块、状态最优估算值计算模块、状态协方差更新模块、测量过程中的噪声生成模块、状态转移过程中产生的噪声生成模块、过程噪声协方差生成模块和测量噪声协方差生成模块;状态转移计算模块、观察测量计算模块、状态预测生成模块、预测协方差生成模块和卡尔曼增益计算模块均与传输矩阵生成模块和观察矩阵生成模块连接;状态最优估算值计算模块和状态协方差更新模块均与观察矩阵生成模块连接;预测协方差生成模块、卡尔曼增益计算模块、测量过程中的噪声生成模块和状态转移过程中产生的噪声生成模块均与过程噪声协方差生成模块和测量噪声协方差生成模块连接;测量过程中的噪声生成模块和状态转移过程中产生的噪声生成模块分别与观察测量计算模块和状态转移计算模块连接。

5.如权利要求4所述的并行卡尔曼滤波系统,其特征在于:

(1)传输矩阵生成模块能够完成步骤(A)中功能;

(2)观察矩阵生成模块能够完成步骤(B)中功能;

(3)状态转移计算模块包括并行的多个独立的子模块,每一个计算状态转移的子模块均能够完成步骤(C)中的完整功能X(k)=A*X(k-1)+L*W(k-1);

(4)观察测量计算模块包括并行的多个独立的子模块,每一个观察测量计算子模块均能够完成步骤(D)中的完整功能Z(k)=H*X(k)+V(k);

(5)状态预测生成模块包括并行的多个独立的子模块,每一个状态预测生成子模块都能够完成步骤(E)中的完整功能X_pre(k)=A*Xkf(k-1);

(6)预测协方差生成模块包括并行的多个独立的子模块,每一个预测协方差生成子模块都能够完成步骤(F)中的完整功能P_pre(k)=A*P(k-1)*A'+Q;

(7)卡尔曼增益计算模块包括并行的多个独立的子模块,每一个卡尔曼增益计算子模块都能够完成步骤(G)中的完整功能Kg(k)=P_pre(k)*H'*inv(H*P_pre(k)*H'+R);

(8)状态最优估算值计算模块包括并行的多个独立的子模块，每一个状态最优估算值计算子模块都能够完成步骤(H)中的完整功能Xkf(k)＝X_pre(k)+Kg(k)*(Z(k)-H*X_pre(k)；

(9)状态协方差更新模块包括并行的多个独立的子模块，每一个状态协方差更新子模块都能够完成步骤(I)中的完整功能P(k)＝(I-Kg(k)*H)*P_pre(k)；

(10)测量过程中的噪声生成模块包括并行的多个独立的子模块，每一个测量过程中噪声生成子模块都能够完成步骤(J)中的完整功能；

(11)状态转移过程中产生的噪声生成模块包括并行的多个独立的子模块，每一个传输噪声生成子模块都能够完成步骤(K)中的完整功能。

6.如权利要求5所述的并行卡尔曼滤波系统，其特征在于：各模块或子模块为ASIC功能模块、一个或多个CPU/GPU核，或者一个或多个FPGA单元。

7.一种并行卡尔曼滤波子模块数据传递及运算方法，其特征在于：包括以下步骤：

1)AM计算生成的传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$ 通过路径1_1，1_2，…，1_n-1，1_n分别传送给SESM1，SESM2，…，SESMn-1，SESMn；

2)AM计算生成的传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$ 通过路径2_1，2_2，…，2_n-1，2_n分别传送给SPESM1，SPESM2，…，SPESMn-1，SPESMn；

3)AM计算生成的传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$ 通过路径3_1，3_2，…，3_n-1，3_n分别传送给CPESM1，CPESM2，…，CPESMn-1，CPESMn；

4)HM计算生成的观察矩阵 $H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$ 通过路径4_1，4_2，…，4_n-1，4_n分别传送给OESM1，OESM2，…，OESMn-1，OESMn；

5)HM计算生成的观察矩阵 $H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$ 通过路径5_1，5_2，…，5_n-1，5_n分别传送给KGESM1，KGESM2，…，KGESMn-1，KGESMn；

6)SESM1计算生成的X(1)＝系统初始状态通过路径6_1传送给SESM2，SESM2计算生成的 $X(2)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_2$ 通过路径6_2传送给SESM3，SESMn-1计算生成的 $X(n-1)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_{n-1}$ 通过路径6_n-1传送给SESMn，SESMn计算生 $X(n)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_n$ 通过路径6_n传送给SESM1，当N<K时，进行此操作；

当N>＝K时，SESMk计算生成 $X(k)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_k$ 后结束操作；

7)WGSM1计算生成的 $W(1)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_1$ 通过路径7_1传送给SESM2，WGSM2计算生成的 $W(2)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_2$ 通过路径7_2传送给SESM3，WGSMn-1计算生 $W(n-1)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_{n-1}$ 通过路径7_n-1传送给SESMn，WGSMn计算生 $W(n)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_n$ 通过路径7_n传送给SESM1，当N<K时，进行此操作；当N>＝K时，WGSMk计算生 $W(k)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_k$ 后结束操作；

8)SESM1计算生 $X(1)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_1$ 通过路径8_1传送给OESM1，SESM2计算生 $X(2)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_2$ 通过路径8_2传送给OESM2，SESMn-1计算生 $X(n-1)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_{n-1}$ 通过路径8_n-1传送给OESMn-1，SESMn计算生 $X(n)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_n$ 通过路径8_n传送给OESMn；

9)VGSM1计算生成的 $V(1)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_1$ 通过路径9_1传送给OESM1，VGSM2计算生成的 $V(2)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_2$ 通过路径9_2传送给OESM2，VGSMn-1计算生成的 $V(n-1)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_{n-1}$ 通过路径9_n-1传送给OESMn-1，VGSMn计算生成的 $V(n)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_n$ 通过路径9_n传送给OESMn；

10)SUESM1计算生成的 $Xkf(1)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_1$ 通过路径10_1传送给SPESM2，SUESM2计算生成的 $Xkf(2)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_2$ 通过路径10_2传送给SPESM3，SUESMn-1计算生成的

$Xkf(n\text{-}1)=\begin{bmatrix}Xkf1\\ \vdots\\ Xkfn\end{bmatrix}_{n-1}$ 通过路径10_n-1传送给SPESMn，SUESMn计算生成的 $Xkf(n)=\begin{bmatrix}Xkf1\\ \vdots\\ Xkfn\end{bmatrix}_{n}$ 通过路径10_n传送给SPESM1，当N<K时，进行此操作；当N>＝K时，SUESMk计算生 $W(k)=\begin{bmatrix}w1\\ \vdots\\ wn\end{bmatrix}_{k}$ 便结束操作；

11)CUESM1计算生成的 $P(1)=\begin{bmatrix}P11 & \cdots & P1n\\ \vdots & \ddots & \vdots\\ Pn1 & \cdots & Pnn\end{bmatrix}_{1}$ 通过路径11_1传送给CPESM2，CUESM2计算生 $P(2)=\begin{bmatrix}P11 & \cdots & P1n\\ \vdots & \ddots & \vdots\\ Pn1 & \cdots & Pnn\end{bmatrix}_{2}$ 通过路径11_2传送给CPESM3，CUESMn-1计算生成的 $P(n\text{-}1)=\begin{bmatrix}P11 & \cdots & P1n\\ \vdots & \ddots & \vdots\\ Pn1 & \cdots & Pnn\end{bmatrix}_{n-1}$ 通过路径11_n-1传送给CPESMn，CUESMn计算生成的 $P(n)=\begin{bmatrix}P11 & \cdots & P1n\\ \vdots & \ddots & \vdots\\ Pn1 & \cdots & Pnn\end{bmatrix}_{n}$ 通过路径11_n传送给CPESM1，当N<K时，进行此操作；当N>＝K时，CUESMk计算生 $P(k)=\begin{bmatrix}P11 & \cdots & P1n\\ \vdots & \ddots & \vdots\\ Pn1 & \cdots & Pnn\end{bmatrix}_{k}$ 后结束操作；

12)QGM计算生成的过程噪声协方差矩阵 $Q=\begin{bmatrix}q11 & \cdots & q1n\\ \vdots & \ddots & \vdots\\ qn1 & \cdots & qnn\end{bmatrix}$ 通过路径12_1，12_2，…，12_n-1，12_n分别传送给CPESM1，CPESM2，…，CPESMn-1，CPESMn；

13)CPESM1计算生成的P_pre(1)＝系统预测协方差初始值通过路径13_1传送给KGESM1，CPESM2计算生成的 $P_pre(2)=\begin{bmatrix}P_pre11 & \cdots & P_pre1n\\ \vdots & \ddots & \vdots\\ P_pren1 & \cdots & P_prenn\end{bmatrix}_{2}$ 通过路径13_2传送给KGESM2，CPESMn-1计算生成的 $P_pre(n\text{-}1)=\begin{bmatrix}P_pre11 & \cdots & P_pre1n\\ \vdots & \ddots & \vdots\\ P_pren1 & \cdots & P_prenn\end{bmatrix}_{n-1}$ 通过路径13_n-1传送给KGESMn-1，CPESMn计算生成的 $P_pre(n)=\begin{bmatrix}P_pre11 & \cdots & P_pre1n\\ \vdots & \ddots & \vdots\\ P_pren1 & \cdots & P_prenn\end{bmatrix}_{n}$ 通过路径13_n传送给

KGESMn；

14)RGM计算生成的测量协方差R通过路径14_1，14_2，…，14_n-1，14_n分别传送给KGESM1，KGESM2，…，KGESMn-1，KGESMn；

15)SPESM1计算生成的 $X_pre(1)=\begin{bmatrix}X_pre1\\ \vdots\\ X_pren\end{bmatrix}_1$ 通过路径15_1传送给SUESM1，SPESM2计算生成的 $X_pre(2)=\begin{bmatrix}X_pre1\\ \vdots\\ X_pren\end{bmatrix}_2$ 通过路径15_2传送给SUESM2，SPESMn-1计算生成的 $X_pre(n\text{-}1)=\begin{bmatrix}X_pre1\\ \vdots\\ X_pren\end{bmatrix}_{n-1}$ 通过路径15_n-1传送给SUESMn-1，SPESMn计算生成的 $X_pre(n)=\begin{bmatrix}X_pre1\\ \vdots\\ X_pren\end{bmatrix}_n$ 通过路径15_n传送给SUESMn；

16)KGESM1计算生成的 $Kg(1)=\begin{bmatrix}Kg11 & \cdots & Kg1n\\ \vdots & \ddots & \vdots\\ Kgn1 & \cdots & Kgnn\end{bmatrix}_1$ 通过路径16_1传送给SUESM1，KGESM2计算生成的 $Kg(2)=\begin{bmatrix}Kg11 & \cdots & Kg1n\\ \vdots & \ddots & \vdots\\ Kgn1 & \cdots & Kgnn\end{bmatrix}_2$ 通过路径16_2传送给SUESM2，KGESMn-1计算生成的 $Kg(n-1)=\begin{bmatrix}Kg11 & \cdots & Kg1n\\ \vdots & \ddots & \vdots\\ Kgn1 & \cdots & Kgnn\end{bmatrix}_{n-1}$ 通过路径16_n-1传送给SUESMn-1，KGESMn计算生成的 $Kg(n)=\begin{bmatrix}Kg11 & \cdots & Kg1n\\ \vdots & \ddots & \vdots\\ Kgn1 & \cdots & Kgnn\end{bmatrix}_n$ 通过路径16_n传送给SUESMn；

17)OESM1计算生成的 $Z(1)=\begin{bmatrix}z1\\ \vdots\\ zn\end{bmatrix}_1$ 通过路径17_1传送给SUESM1，OESM2计算生成的 $Z(2)=\begin{bmatrix}z1\\ \vdots\\ zn\end{bmatrix}_2$ 通过路径17_2传送给SUESM2，OESMn-1计算生成的 $Z(n\text{-}1)=\begin{bmatrix}z1\\ \vdots\\ zn\end{bmatrix}_{n-1}$ 通过路径17_n-1传送给SUESMn-1，OESMn计算生成的 $Z(n)=\begin{bmatrix}z1\\ \vdots\\ zn\end{bmatrix}_n$ 通过路径17_n传送给SUESMn；

18)KGESM1计算生成的 $Kg(1)=\begin{bmatrix}Kg11 & \cdots & Kg1n\\ \vdots & \ddots & \vdots\\ Kgn1 & \cdots & Kgnn\end{bmatrix}_1$ 通过路径18_1传送给CUESM1，

KGESM2计算生成的 $Kg(2)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_2$ 通过路径18_2传送给CUESM2，KGESMn-1计算生成的 $Kg(n-1)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_{n-1}$ 通过路径18_n-1传送给CUESMn-1，KGESMn计算生成的 $Kg(n)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_n$ 通过路径18_n传送给CUESMn；

19)CPESM1计算生成的 $P_pre(1)=\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_1$ 通过路径19_1传送给CUESM1，CPESM2计算生成的 $P_pre(2)=\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_2$ 通过路径19_2传送给CUESM2，CPESMn-1计算生成的 $P_pre(n-1)=\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_{n-1}$ 通过路径19_n-1传送给CUESMn-1，CPESMn计算生成的 $P_pre(n)=\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_n$ 通过路径19_n传送给CUESMn；

20)RGM计算生成的测量噪声协方差R通过路径20_1，20_2，…，20_n-1，20_n分别传送给VGSM1，VGSM2，…，VGSMn-1，VGSMn；

21)QGM计算生成的过程噪声协方差矩阵 $Q=\begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix}$ 通过路径21_1，21_2，…，21_n-1，21_n分别传送给WGSM1，WGSM2，…，WGSMn-1，WGSMn。

8.如权利要求7所述的并行卡尔曼滤波子模块数据传递及运算方法，其特征在于：载波追踪的状态点的个数为20000个，选用多普勒频偏DetaTheta，频偏的变化率Omega0，频偏变化率的变化率Omega1作为系统的状态量，系统为三阶系统，detav＝1e-3，detaw＝1e-3，Ts＝1e-3，$L=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$；HM计算观测矩阵的经验公式为H＝[1 Ts/2 Ts^2/6]＝[1 5.0e-04 1.67e-07]；AM计算观测矩阵的经验公式为

$$A=\begin{bmatrix} 1 & Ts & Ts\text{^}2/2 \\ 0 & 1 & Ts \\ 0 & 0 & 1 \end{bmatrix}=\begin{bmatrix} 1 & 1.0e-03 & 5.0e-07 \\ 0 & 1 & 1.0e-03 \\ 0 & 0 & 1 \end{bmatrix};$$

QGM计算过程噪声协方差矩阵

的经验公式为Q=detav^2* $\begin{bmatrix} Ts^6/252 & Ts^5/72 & Ts^4/30 \\ Ts^5/72 & Ts^4/20 & Ts^3/8 \\ Ts^4/30 & Ts^3/8 & Ts^2/3 \end{bmatrix}$ = $\begin{bmatrix} 3.9689e-27 & 1.39e-23 & 3.33e-20 \\ 1.39e-23 & 5.00e-20 & 1.25e-16 \\ 3.33e-20 & 1.25e-16 & 3.33e-13 \end{bmatrix}$; RGM计算测量噪声协方差的经验公式为R＝detaw^2/Ts^4＝1.0e+06;VGSM计算测量噪声V＝N(0,R);WGSM计算传输噪声W＝N(0,Q);设定系统状态量的初始值为X(1)＝[0 -20*e3 0],系统观测量的初始值为Z(1)＝[0 0 0],系统状态协方差初始值 $P(1)=\begin{bmatrix} 1/12 & 0 & 0 \\ 0 & 1*10^{(-4)} & 0 \\ 0 & 0 & 1*10^{(-4)} \end{bmatrix}$, 预测协方差初始值 $P_pre(1)=\begin{bmatrix} 1/12 & 0 & 0 \\ 0 & 1*10^{(-4)} & 0 \\ 0 & 0 & 1*10^{(-4)} \end{bmatrix}$, 状态最优估算值的初始值Xkf(1)＝[0 -20*10^(3) 0],状态预测值的初始值X_pre(1)＝[0 0 0]。

9.如权利要求8所述的并行卡尔曼滤波子模块数据传递及运算方法,其特征在于:n为500,则,SESMk完成X(k)＝A*X(k-1)+L*W(k-1);

(A)OESMk完成Z(k)＝H*X(k)+V(k);

(C)CPESMk完成P_pre(k)＝A*P(k-1)*A'+Q;

(D)KGESMk完成Kg(k)＝P_pre(k)*H'*inv(H*P_pre(k)*H'+R);

(E)SUESMk完成Xkf(k)＝X_pre(k)+Kg(k)*(Z(k)-H*X_pre(k));

(F)CUESMk完成P(k)＝(I-Kg(k)*H)*P_pre(k)。

10.如权利要求9所述的并行卡尔曼滤波子模块数据传递及运算方法,其特征在于:包括以下步骤,

1)完成6轮运算,产生3000个系统状态值;

2)设定系统状态量的初始值Omega1即X(3,3000)＝80*10^(3),状态最优估算值的初始值Xkf(3,300)＝80*10^(3),其他数值保持不变,完成1轮运算,产生500个系统状态值;

3)设定系统状态量的初始值Omega1即X(3,3500)＝0,状态最优估算值的初始值Xkf(3,3500)＝0,其他数值保持不变,完成11轮运算,产生5500个系统状态值;

4)设定系统状态量的初始值Omega1即X(3,9000)＝-80*10^(3),状态最优估算值的初始值Xkf(3,3500)＝-80*10^(3),其他数值保持不变,完成1轮运算,产生500个系统状态值;

5)设定系统状态量的初始值Omega1即X(3,9500)＝0,状态最优估算值的初始值Xkf(3,3500)＝0,其他数值保持不变,完成12轮运算,产生6000个系统状态值;

6)设定系统状态量的初始值Omega1即X(3,15500)＝80*10^(3),状态最优估算值的初始值Xkf(3,3500)＝80*10^(3),其他数值保持不变,完成1轮运算,产生500个系统状态值;

7)设定系统状态量的初始值Omega1即X(3,16000)＝0,状态最优估算值的初始值Xkf(3,3500)＝0,其他数值保持不变,完成8轮运算,产生4000个系统状态值。

并行卡尔曼滤波系统和方法

技术领域

[0001] 本发明涉及通讯技术领域，具体涉及一种通讯技术领域的数字信号并行卡 尔曼滤波方法。

背景技术

[0002] 滤波理论是在对系统可观察信号进行测量的基础上，根据一定的滤波准则，对系统的状态进行估计的理论和方法。根据贝叶斯理论，最优估计只是一个理 想化的解，一般情况下，无法得到它的解析形式。20世纪60年代，由R.E.Kalman 提出了卡尔曼滤波方法，在时域内以递推方式进行估计，其滤波标准是均方误 差最小，当系统为线性高斯分布时，它可以得到递归贝叶斯估计的最小均方误 差解，此方法克服了经典维纳滤波算法的局限，是最优化的自回归数据处理算 法。也是目前最经典、应用最广泛的线性滤波方法。此算法选用状态空间方法 来设计滤波器，能对多维系统和非平稳系统的随机过程进行估计。卡尔曼滤波 器因其自身的优点：递归运算、自适应性、前瞻性等，能够对随机干扰下的线 性动态系统进行最优估计，因而有着广泛的应用。卡尔曼滤波已经被应用于航 空航天、通信工程、人工智能、图像处理、轨道交通、石油勘探等许多领域。但传统的卡尔曼滤波方法，运行效率低，占用了较多的计算资源，无法同时实 现多个状态的卡尔曼滤波。

发明内容

[0003] 本发明要解决的技术问题是提供一种并行卡尔曼滤波方法，其相对于传统 的卡尔曼滤波方法，能够有效的提升卡尔曼滤波的运行效率，可以同时实现多 个状态的卡尔曼滤波，也兼顾了计算效率与计算资源之间的平衡问题。

[0004] 为了解决上述技术问题，本发明提供了一种并行卡尔曼滤波方法，包括以 下步骤，将卡尔曼的运算流程拆分为多个步骤，在使每个步骤之间进行并行运 算的同时，也使每个步骤用多个并行的子模块进行同时运算。

[0005] 本发明一个较佳实施例中，进一步包括卡尔曼运算流程至少拆分成以下步 骤：S1，生成传输矩阵A；S2，生成观察矩阵H；S3，计算系统的状态转移； S4，计算观察测量值；S5，生成状态预测；S6，生成预测协方差P_pre(k)；S7， 计算卡尔曼增益；S8，计算状态最优估算值；S9，更新状态协方差；S10，产 生测量过程中遇到的噪声V和状态转移过程中产生的噪声W。

[0006] 本发明一个较佳实施例中，进一步包括以下步骤，

[0007] (A)计算传输矩阵A：取A为n*n维矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$；

[0008] (B)计算观察矩阵H：取H为n*n维矩阵 $H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$；

[0009] (C)计算系统的状态值：由状态转移方程X(k)＝A*X(k-1)+L*W(k-1)计算 系统的

状态转移；

[0010] (D)计算观察测量值：采用步骤(B)中计算出的矩阵 $H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$ 和第k个状态矩阵 $X(K)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_k$ 相乘再和第k个状态下的测量噪声矩阵V(K) 相加得到第k个状态下的测量值Z(K)，观察矩阵的计算方程为：Z(k)＝H*X(k)+V(k)；

[0011] (E)生成状态预测：采用步骤(A)计算传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$ 和 步骤(H)计算出的第k-1个状态的最优估计值Xkf(k-1)，做矩阵乘法得到第k 个状态的预测值X_pre(k)，取Xkf(k-1)为n维列向量 $Xkf(k\text{-}1)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_{k-1}$，取X_pre(k)为n维列向量 $X_pre(k)=\begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_k$；状态预测的计算方程为：X_pre(k)＝A*Xkf(k-1)；

[0012] (F)生成预测协方差P_pre(k)：取P_pre(k)为n*n维矩阵 $P_pre(k)=\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k$，其计算方程为P_pre(k)＝A*P(k-1)*A'+Q；

[0013] (G)计算卡尔曼增益：取卡尔曼增益Kg为n维列向量 $Kg=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}$，其计算方程为 Kg(k)＝P_pre(k)*H'*inv(H*P_pre(k)*H'+R)；

[0014] (H)计算状态最优估算值：取状态最优估算值 $Xkf(k)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_k$，其计 算方程为Xkf(k)＝X_pre(k)+Kg(k)*(Z(k)-H*X_pre(k))；

[0015] (I)更新状态协方差：其计算方程为P(k)＝(I-Kg(k)*H)*P_pre(k)；

[0016] (J)(K)分别为是为了产生测量过程中遇到的噪声和状态转移过程中产 生的噪声 $V(K)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_k$ 和 $W(K)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_k$，这两个噪声的协方差分别为R和 $Q=\begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix}$。

[0017] 一种并行卡尔曼滤波系统，包括传输矩阵生成模块、观察矩阵生成模块、状态转

移计算模块、观察测量计算模块、状态预测生成模块、预测协方差生成 模块、卡尔曼增益计算模块、状态最优估算值计算模块、状态协方差更新模块、 测量过程中的噪声生成模块、状态转移过程中产生的噪声生成模块、过程噪声 协方差生成模块和测量噪声协方差生成模块;状态转移计算模块、观察测量计 算模块、状态预测生成模块、预测协方差生成模块和卡尔曼增益计算模块均与 传输矩阵生成模块和观察矩阵生成模块连接;状态最优估算值计算模块和状态 协方差更新模块均与观察矩阵生成模块连接;预测协方差生成模块、卡尔曼增 益计算模块、测量过程中的噪声生成模块和状态转移过程中产生的噪声生成模 块均与过程噪声协方差生成模块和测量噪声协方差生成模块连接;测量过程中 的噪声生成模块和状态转移过程中产生的噪声生成模块分别与观察测量计算模 块和状态转移计算模块连接。

[0018] 本发明一个较佳实施例中,进一步包括(1)传输矩阵生成模块能够完成步 骤(A)中功能;

[0019] (2)观察矩阵生成模块能够完成步骤(B)中功能;

[0020] (3)状态转移计算模块包括并行的多个独立的子模块,每一个计算状态转 移的子模块均能够完成步骤(C)中的完整功能X(k)＝A*X(k-1)+L*W(k-1);

[0021] (4)观察测量计算模块包括并行的多个独立的子模块,每一个观察测量计 算子模块均能够完成步骤(D)中的完整功能Z(k)＝H*X(k)+V(k);

[0022] (5)状态预测生成模块包括并行的多个独立的子模块,每一个状态预测生 成子模块都能够完成步骤(E)中的完整功能X_pre(k)＝A*Xkf(k-1);

[0023] (6)预测协方差生成模块包括并行的多个独立的子模块,每一个预测协方 差生成子模块都能够完成步骤(F)中的完整功能P_pre(k)＝A*P(k-1)*A'+Q;

[0024] (7)卡尔曼增益计算模块包括并行的多个独立的子模块,每一个卡尔曼增 益计算子模块都能够完成步骤(G)中的完整功能Kg(k)＝P_pre(k)*H'*inv(H*P_pre(k)*H'+R);

[0025] (8)状态最优估算值计算模块包括并行的多个独立的子模块,每一个状态 最优估算值计算子模块都能够完成步骤(H)中的完整功能 Xkf(k)＝X_pre(k)+Kg(k)*(Z(k)-H*X_pre(k);

[0026] (9)状态协方差更新模块包括并行的多个独立的子模块,每一个状态协方 差更新子模块都能够完成步骤(I)中的完整功能P(k)＝(I-Kg(k)*H)*P_pre(k);

[0027] (10)测量过程中的噪声生成模块包括并行的多个独立的子模块,每一个 测量过程中噪声生成子模块都能够完成步骤(J)中的完整功能;

[0028] (11)状态转移过程中产生的噪声生成模块包括并行的多个独立的子模块, 每一个传输噪声生成子模块都能够完成步骤(K)中的完整功能。

[0029] 本发明一个较佳实施例中,进一步包括各模块或子模块为ASIC功能模块、一个或多个CPU/GPU核,或者一个或多个FPGA单元。

[0030] 一种并行卡尔曼滤波子模块数据传递及运算方法,包括以下步骤:

[0031] 1)AM计算生成的传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$ 通过路径1_1,1_2,…,1_n-1, 1_n分别传送给SESM1,SESM2,…,SESMn-1,SESMn;

[0032] 2)AM计算生成的传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$ 通过路径2_1,2_2,…,2_n-1，2_n分别传送给SPESM1,SPESM2,…,SPESMn-1,SPESMn；

[0033] 3)AM计算生成的传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$ 通过路径3_1,3_2,…,3_n-1，3_n分别传送给CPESM1,CPESM2,…,CPESMn-1,CPESMn；

[0034] 4)HM计算生成的观察矩阵 $H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$ 通过路径4_1,4_2,…,4_n-1，4_n分别传送给OESM1,OESM2,…,OESMn-1,OESMn；

[0035] 5)HM计算生成的观察矩阵 $H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$ 通过路径5_1,5_2,…,5_n-1，5_n分别传送给KGESM1,KGESM2,…,KGESMn-1,KGESMn；

[0036] 6)SESM1计算生成的X(1)＝系统初始状态通过路径6_1传送给SESM2，SESM2计算生成的 $X(2)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_2$ 通过路径6_2传送给SESM3,SESMn-1计算生 成的 $X(n-1)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_{n-1}$ 通过路径6_n-1传送给SESMn,SESMn计算生 $X(n)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_n$ 通 过路径6_n传送给SESM1,当N<K时,进行此操作;当N>＝K时,SESMk计 算生成 $X(k)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_k$ 后结束操作；

[0037] 7)WGSM1计算生成的 $W(1)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_1$ 通过路径7_1传送给SESM2,WGSM2 计算生 $W(2)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_2$ 通过路径7_2传送给SESM3,WGSMn-1计算生 $W(n-1)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_{n-1}$ 通过路径7_n-1传送给SESMn,WGSMn计算生 $W(n)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_n$ 通过 路径7_n传送给SESM1,当N<K时,进行此操作;当N>＝K时,WGSMk计 算生 $W(k)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_k$ 后结束操作；

[0038] 8)SESM1计算生 $X(1)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_1$ 通过路径8_1传送给OESM1,SESM2计算生 $X(2)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_2$

通过路径8_2传送给OESM2，SESMn-1计算生 $X(n-1)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_{n-1}$ 通过路 径8_n-1传送给OESMn-1，SESMn计算生 $X(n)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_{n}$ 通过路径8_n传送给 OESMn；

[0039] 9)VGSM1计算生成的 $V(1)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_{1}$ 通过路径9_1传送给OESM1，VGSM2 计算生成的 $V(2)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_{2}$ 通过路径9_2传送给OESM2，VGSMn-1计算生成的 $V(n-1)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_{n-1}$ 通过路径9_n-1传送给OESMn-1，VGSMn计算生成的 $V(n)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_{n}$ 通过路径9_n传送给OESMn；

[0040] 10)SUESM1计算生成的 $Xkf(1)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_{1}$ 通过路径10_1传送给SPESM2，SUESM2计算生成的 $Xkf(2)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_{2}$ 通过路径10_2传送给SPESM3，SUESMn-1 计算生成的 $Xkf(n-1)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_{n-1}$ 通过路径10_n-1传送给SPESMn，SUESMn计 算生成的 $Xkf(n)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_{n}$ 通过路径10_n传送给SPESM1，当N<K时，进行此 操作；当N>＝K时，SUESMk计算生 $W(k)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_{k}$ 便结束操作；

[0041] 11)CUESM1计算生成的 $P(1)=\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_{1}$ 通过路径11_1传送给CPESM2，CUESM2计算生 $P(2)=\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_{2}$ 通过路径11_2传送给CPESM3，CUESMn-1计算生成的 $P(n-1)=\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_{n-1}$ 通过路径11_n-1传送给 CPESMn，CUESMn计算生成的

$P(n)=\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_n$ 通过路径11_n传送给 CPESM1，当N<K时，进行此操作；当N>＝K时，CUESMk计算生 $P(k)=\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_k$ 后结束操作；

[0042] 12)QGM计算生成的过程噪声协方差矩阵 $Q=\begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix}$ 通过路径12_1，12_2，…，12_n-1，12_n分别传送给CPESM1，CPESM2，…，CPESMn-1，CPESMn；

[0043] 13)CPESM1计算生成的P_pre(1)＝系统预测协方差初始值通过路径13_1 传送给KGESM1，CPESM2计算生成的 $P_pre(2)=\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_2$ 通过路径 13_2传送给KGESM2，CPESMn-1计算生成的 $P_pre(n-1)=\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_{n-1}$ 通过路径13_n-1传送给KGESMn-1，CPESMn计算生成的 $P_pre(n)=\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_n$ 通过路径13_n传送给KGESMn；

[0044] 14)RGM计算生成的测量协方差R通过路径14_1，14_2，…，14_n-1，14_n分别传送给KGESM1，KGESM2，…，KGESMn-1，KGESMn；

[0045] 15)SPESM1计算生成的 $X_pre(1)=\begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_1$ 通过路径15_1传送给 SUESM1，SPESM2计算生成的 $X_pre(2)=\begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_2$ 通过路径15_2传送给 SUESM2，SPESMn-1计算生成的 $X_pre(n-1)=\begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_{n-1}$ 通过路径15_n-1传送 给SUESMn-1，SPESMn计算生成的 $X_pre(n)=\begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_n$ 通过路径15_n传送给 SUESMn；

[0046] 16)KGESM1计算生成的 $Kg(1)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_1$ 通过路径16_1传送给

SUESM1，KGESM2计算生成的Kg$(2)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_2$通过路径16_2传送 给SUESM2，KGESMn-1计算生成的Kg$(n-1)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_{n-1}$通过路径 16_n-1传送给SUESMn-1，KGESMn计算生成的Kg$(n)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_n$通 过路径16_n传送给SUESMn；

[0047] 17)OESM1计算生成的 $Z(1)=\begin{bmatrix} z1 \\ \vdots \\ zn \end{bmatrix}_1$通过路径17_1传送给SUESM1，OESM2 计算生成的 $Z(2)=\begin{bmatrix} z1 \\ \vdots \\ zn \end{bmatrix}_2$通过路径17_2传送给SUESM2，OESMn-1计算生成的 $Z(n-1)=\begin{bmatrix} z1 \\ \vdots \\ zn \end{bmatrix}_{n-1}$通过路径17_n-1传送给SUESMn-1，OESMn计算生成的 $Z(n)=\begin{bmatrix} z1 \\ \vdots \\ zn \end{bmatrix}_n$通过路径17_n传送给SUESMn；

[0048] 18)KGESM1计算生成的Kg$(1)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_1$通过路径18_1传送给CUESM1，KGESM2计算生成的Kg$(2)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_2$通过路径18_2传送 给CUESM2，KGESMn-1计算生成的Kg(n-1)Kg$(n-1)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_{n-1}$通过路径18_n-1传送给CUESMn-1，KGESMn计算生成的 Kg$(n)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_n$通过路径18_n传送给CUESMn；

[0049] 19)CPESM1计算生成的 $P_pre(1)=\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_1$通过路径19_1传送 给CUESM1，CPESM2计算生成的 $P_pre(2)=\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_2$通过路径19_2 传送给

CUESM2,CPESMn-1计算生成的 $P_pre(n-1)=\begin{bmatrix} P_{_pre11} & \cdots & P_{_pre1n} \\ \vdots & \ddots & \vdots \\ P_{_pren1} & \cdots & P_{_prenn} \end{bmatrix}_{n-1}$ 通过 路径19_n-1传送给CUESMn-1,CPESMn计算生成的 $P_pre(n)=\begin{bmatrix} P_{_pre11} & \cdots & P_{_pre1n} \\ \vdots & \ddots & \vdots \\ P_{_pren1} & \cdots & P_{_prenn} \end{bmatrix}_{n}$ 通过路径19_n传送给CUESMn;

[0050] 20)RGM计算生成的测量噪声协方差R通过路径20_1,20_2,…,20_n-1, 20_n分别传送给VGSM1,VGSM2,…,VGSMn-1,VGSMn;

[0051] 21)QGM计算生成的过程噪声协方差矩阵 $Q=\begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix}$ 通过路径21_1, 21_2,…,21_n-1,21_n分别传送给WGSM1,WGSM2,…,WGSMn-1,WGSMn。

[0052] 本发明一个较佳实施例中,进一步包括载波追踪的状态点的个数为20000 个,选用多普勒频偏DetaTheta,频偏的变化率Omega0,频偏变化率的变化率 Omega1作为系统的状态量,系统为三阶系统,detav=1e-3,detaw=1e-3,Ts=1e-3, $L=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$; HM计算观测矩阵的经验公式为 H=[1 Ts/2 Ts^2/6]=[1 5.0e-04 1.67e-07];AM计算观测矩阵的经验 公式为 $A=\begin{bmatrix} 1 & Ts & Ts^2/2 \\ 0 & 1 & Ts \\ 0 & 0 & 1 \end{bmatrix}=\begin{bmatrix} 1 & 1.0e-03 & 5.0e-07 \\ 0 & 1 & 1.0e-03 \\ 0 & 0 & 1 \end{bmatrix}$; QGM计算过程噪声 协方差矩阵的经验公式为 $Q=detav^2*\begin{bmatrix} Ts^6/252 & Ts^5/72 & Ts^4/30 \\ Ts^5/72 & Ts^4/20 & Ts^3/8 \\ Ts^4/30 & Ts^3/8 & Ts^2/3 \end{bmatrix}=$

[0053] $\begin{bmatrix} 3.9689e-27 & 1.39e-23 & 3.33e-20 \\ 1.39e-23 & 5.00e-20 & 1.25e-16 \\ 3.33e-20 & 1.25e-16 & 3.33e-13 \end{bmatrix}$; RGM计算测量噪声协方差的经验公式为R=detaw^2/Ts^4=1.0e+06;VGSM计算测量噪声V=N(0,R);WGSM计算 传输噪声W=N(0,Q);设定系统状态量的初始值为X(1)=[0 -20*e3 0], 系统观测量的初始值为Z(1)=[0 0 0],系统状态协方差初始值 $P(1)=\begin{bmatrix} 1/12 & 0 & 0 \\ 0 & 1*10^{(-4)} & 0 \\ 0 & 0 & 1*10^{(-4)} \end{bmatrix}$, 预测协方差初始值 $P_pre(1)=\begin{bmatrix} 1/12 & 0 & 0 \\ 0 & 1*10^{(-4)} & 0 \\ 0 & 0 & 1*10^{(-4)} \end{bmatrix}$, 状态最优估算值的初始值Xkf (1)=[0 -20*10^(3) 0],状态预测值的初始值X_pre(1)=[0 0 0]。

[0054] 本发明一个较佳实施例中,进一步包括n为500,则,SESMk完成 X(k)=A*X(k-1)+L*W(k-1);

[0055] (A)OESMk完成Z(k)＝H*X(k)+V(k)；

[0056] (C)CPESMk完成P_pre(k)＝A*P(k-1)*A'+Q；

[0057] (D)KGESMk完成Kg(k)＝P_pre(k)*H'*inv(H*P_pre(k)*H'+R)；

[0058] (E)SUESMk完成Xkf(k)＝X_pre(k)+Kg(k)*(Z(k)-H*X_pre(k))；

[0059] (F)CUESMk完成P(k)＝(I-Kg(k)*H)*P_pre(k)。

[0060] 本发明一个较佳实施例中，进一步包括以下步骤，

[0061] 1)完成6轮运算，产生3000个系统状态值；

[0062] 2)设定系统状态量的初始值Omega1即X(3,3000)＝80*10^(3)，状态最优估 算值的初始值Xkf(3,300)＝80*10^(3)，其他数值保持不变，完成1轮运算，产 生500个系统状态值；

[0063] 3)设定系统状态量的初始值Omega1即X(3,3500)＝0，状态最优估算值的 初始值Xkf(3,3500)＝0，其他数值保持不变，完成11轮运算，产生5500个系 统状态值；

[0064] 4)设定系统状态量的初始值Omega1即X(3,9000)＝-80*10^(3)，状态最优 估算值的初始值Xkf(3,3500)＝-80*10^(3)，其他数值保持不变，完成1轮运算， 产生500个系统状态值；

[0065] 5)设定系统状态量的初始值Omega1即X(3,9500)＝0，状态最优估算值的 初始值Xkf(3,3500)＝0，其他数值保持不变，完成12轮运算，产生6000个系 统状态值；

[0066] 6)设定系统状态量的初始值Omega1即X(3,15500)＝80*10^(3)，状态最优 估算值的初始值Xkf(3,3500)＝80*10^(3)，其他数值保持不变，完成1轮运算， 产生500个系统状态值；

[0067] 7)设定系统状态量的初始值Omega1即X(3,16000)＝0，状态最优估算值的 初始值Xkf(3,3500)＝0，其他数值保持不变，完成8轮运算，产生4000个系统 状态值。

[0068] 本发明一个较佳实施例中，进一步包括。

[0069] 本发明的有益效果：

[0070] 本发明的并行卡尔曼滤波方法，其相对于传统的卡尔曼滤波方法，能够有 效的提升卡尔曼滤波的运行效率，可以同时实现多个状态的卡尔曼滤波，也兼 顾了计算效率与计算资源之间的平衡问题。

附图说明

[0071] 图1为本发明优选实施例中并行卡尔曼滤波系统的结构示意图；

[0072] 图2为本发明优选实施例中并行卡尔曼滤波系统的局部放大示意图一；

[0073] 图3为本发明优选实施例中并行卡尔曼滤波系统的局部放大示意图二；

[0074] 图4为本发明优选实施例中并行卡尔曼滤波系统的运算原理图；

[0075] 图5为并行卡尔曼滤波子模块间数据传递及运算关系示意图；

[0076] 图6为500个子模块的并行卡尔曼滤波子模块间数据传递及运算关系示意 图；

[0077] 图7为实例计算生成的20000个系统状态中的多普勒频偏DetaTheta；

[0078] 图8为实例计算生成的20000个系统状态中的多普勒频偏的变化率 Omega0；

[0079] 图9为实例计算生成的20000个系统状态中的多普勒频偏变化率的变化率 Omega1；

[0080] 图10为实例计算生成的20000个系统状态的测量错误和卡尔曼估计误差。

具体实施方式

[0081] 下面结合附图和具体实施例对本发明作进一步说明，以使本领域的技术人 员可以更好地理解本发明并能予以实施，但所举实施例不作为对本发明的限定。

[0082] 实施例

[0083] 本发明实施例公开一种并行卡尔曼滤波方法，参照图1～4所示，包括以下 步骤：将卡尔曼的运算流程拆分为多个步骤，在使每个步骤之间进行并行运算 的同时，也使每个步骤用多个并行的子模块进行同时运算。其相对于传统的卡 尔曼滤波方法，能够有效的提升卡尔曼滤波的运行效率，可以同时实现多个状 态的卡尔曼滤波，也兼顾了计算效率与计算资源之间的平衡问题。

[0084] 在拆分卡尔曼运算流程时，至少包括以下步骤：

[0085] S1，生成传输矩阵A；

[0086] S2，生成观察矩阵H；

[0087] S3，计算系统的状态转移；

[0088] S4，计算观察测量值；

[0089] S5，生成状态预测；

[0090] S6，生成预测协方差P_pre(k)；

[0091] S7，计算卡尔曼增益；

[0092] S8，计算状态最优估算值；

[0093] S9，更新状态协方差；

[0094] S10，产生测量过程中遇到的噪声V和状态转移过程中产生的噪声W。

[0095] 具体而言，参照图4所示，包括以下步骤：

[0096] (A)传输矩阵A生成部分用来计算出传输矩阵A(这里我们取A为n*n 维矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$)，系统的状态维度决定着A的维度，A的运算过 程及运算过程中涉及的参数根据系统本身的传输特性并依据过往经验对系统在 状态转移过程中的修正给出。

[0097] (B)观察矩阵H生成部分用来计算出观察矩阵H(这里我们取H为n*n 维矩阵 $H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$)，系统的状态维度决定着H的维度，H的运算过 程及运算过程中涉及的参数根据测量过程(或观察过程)的待观察量并依据过往 经验对测量系统的修正给出。

[0098] (C)状态转移计算部分用来计算系统的状态值，系统的状态转移的计算 过程是根据状态转移方程X(k)＝A*X(k-1)+L*W(k-1)来进行的，其中X(这里我们 取X为n维列向量 $X=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}$)是系统的状态矩阵，k指明该状态是转移过程中 的第k个状态X(k)(这里我们

取X(k)为n维列向量 $X(k)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_k$),k-1指明该 状态是转移过程中的第k-1个状态X(k-1)(这里我们取X(k-1)为n维列向量 $X(k-1)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_{k-1}$)。L(这里我们取L为n*n维矩阵 $L=\begin{bmatrix} l11 & \cdots & l1n \\ \vdots & \ddots & \vdots \\ ln1 & \cdots & lnn \end{bmatrix}$)是调节 状态转移过程中噪声的参数矩阵,其维度由系统的状态维度决定,W(这里我们 取W为n维列向量 $W=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}$)将在(K)部分进行说明。则状态转移方程 X(k)＝A*X(k-1)+L*W(k-1)可以表示成矩阵形式为,

[0099] $\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_k=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}*\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_{k-1}+\begin{bmatrix} l11 & \cdots & l1n \\ \vdots & \ddots & \vdots \\ ln1 & \cdots & lnn \end{bmatrix}*\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_{k-1}$, 即

[0100] $\begin{bmatrix} x1\ (k) \\ \vdots \\ xn\ (k) \end{bmatrix}=$

[0101] $\begin{bmatrix} (a11*x1\ (k-1)+\cdots a1n*xn\ (k-1))+(l11*w1\ (k-1)+\cdots l1n*wn\ (k-1)) \\ \vdots \\ (an1*x1\ (k-1)+\cdots ann*xn\ (k-1))+(ln1*w1\ (k-1)+\cdots lnn*wn\ (k-1)) \end{bmatrix}$。

[0102] (D)观察测量值的计算用(B)中计算出的矩阵 $H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$ 和第 k个状态矩阵 $X(K)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_k$ 相乘再和第k个状态下的测量噪声矩阵V(K) (这里我们取V(K)为n维列向量 $V(K)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_k$)相加得到第k个状态 下的测量值Z(K)(这里我们取Z(K)为n维列向量 $Z(K)=\begin{bmatrix} z1 \\ \vdots \\ zn \end{bmatrix}_k$),观察矩阵 的计算方程为:Z(k)＝H*X(k)+V(k),表示成矩阵形式为,

[0103] $\begin{bmatrix} z1 \\ \vdots \\ zn \end{bmatrix}_k=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}*\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_k+\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_k$, 即

[0104] $\begin{bmatrix} z1\ (k) \\ \vdots \\ zn\ (k) \end{bmatrix}=\begin{bmatrix} (h11*x1\ (k)+\cdots h1n*xn\ (k))+v1\ (k) \\ \vdots \\ (hn1*x1\ (k)+\cdots hnn*xn\ (k))+vn\ (k) \end{bmatrix}$。

[0105] (E)状态预测生成部分用(A)计算出传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$ 和(H) 计算出的第k-1个状态的最优估计值Xkf(k-1)(这里我们取Xkf(k-1)为n维列向 量 $Xkf(k\text{-}1)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_{k-1}$)做矩阵乘法得到第k个状态的预测值X_pre(k)(这里我 们取X_pre(k)为n维列向量 $X_pre(k)=\begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_k$)。状态预测的计算方程为：X_pre(k)＝A*Xkf(k-1),表示成矩阵形式为，

[0106] $\begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_k= \begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}* \begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_{k-1}$, 即

[0107] $\begin{bmatrix} X_pre1\ (k) \\ \vdots \\ X_pren\ (k) \end{bmatrix}=\begin{bmatrix} (a11*Xkf1\ (k-1)\ +\cdots a1n*Xkfn\ (k-1)) \\ \vdots \\ (an1*Xkf1\ (k-1)\ +\cdots ann*Xkfn\ (k-1)) \end{bmatrix}$。

[0108] (F)预测协方差生成部分用来计算预测状态的协方差P_pre(k)(这里我们 取P_pre(k)为n*n维矩阵 $P_pre(k)=\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k$),预测状态的协 方差是用(I)计算出的第k-1个状态的协方差P(k-1)(这里我们取P(k-1)为n*n 维矩阵 $P(k\text{-}1)=\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_{k-1}$)和(A)中计算出的传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$)和(L)中计算出的过程噪声协方差矩阵Q(这里我们取Q 为n*n维矩阵 $Q=\begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix}$)运算产生,其计算方程为 P_pre(k)＝A*P(k-1)*A'+Q,表示成矩阵形式为， $\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k = \begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix} * \begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_{k-1} * \begin{bmatrix} a11 & \cdots & an1 \\ \vdots & \ddots & \vdots \\ a1n & \cdots & ann \end{bmatrix} + \begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix}$。这里设

$\begin{bmatrix} P_p11 & \cdots & P_p1n \\ \vdots & \ddots & \vdots \\ P_pn1 & \cdots & P_pnn \end{bmatrix}_{k-1}= \begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}* \begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_{k-1}=$

$\begin{bmatrix} (a11*p11(k-1)+\cdots+a1n*pn1(k-1)) & \cdots & (a11*p1n(k-1)+\cdots+a1n*pnn(k-1)) \\ \vdots & \ddots & \vdots \\ (an1*p11(k-1)+\cdots ann*pn1(k-1)) & \cdots & (an1*p1n(k-1)+\cdots+ann*pnn(k-1)) \end{bmatrix}$。则

$$\begin{bmatrix} P_{_pre11} & \cdots & P_{_pre1n} \\ \vdots & \ddots & \vdots \\ P_{_pren1} & \cdots & P_{_prenn} \end{bmatrix}_k = \begin{bmatrix} P_{p11} & \cdots & P_{p1n} \\ \vdots & \ddots & \vdots \\ P_{pn1} & \cdots & P_{pnn} \end{bmatrix}_{k-1} * \begin{bmatrix} a11 & \cdots & an1 \\ \vdots & \ddots & \vdots \\ a1n & \cdots & ann \end{bmatrix} + \begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix} =$$

$$\begin{bmatrix} (Pp11(k-1)*a11+\cdots+Pp1n(k-1)*a1n+q11) & \cdots & (Pp11(k-1)*an1+\cdots+Pp1n(k-1)*ann+q1n) \\ \vdots & \ddots & \vdots \\ (Ppn1(k-1)*a11+\cdots+Ppnn(k-1)*a1n+qn1) & \cdots & (Ppn1(k-1)*a1n+\cdots+Ppnn(k-1)*ann+qnn) \end{bmatrix}$$

[0109] (G)卡尔曼增益计算部分是用来计算卡尔曼增益Kg(KalmanGain)(这 里我们取Kg为n维列向量 $Kg = \begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}$)和用(F)计算出的第k 个状态预测协方差 $P_pre(k) = \begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k$ 和(B)计算出来观察矩 阵 $H = \begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$ 和(M)计算出的测量噪声协方差R运算得出。其计算 方程为Kg(k)＝P_pre(k)*H′*inv(H*P_pre(k)*H′+R)。表示成矩阵形式为，

$$\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_k = \begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k * \begin{bmatrix} h11 & \cdots & hn1 \\ \vdots & \ddots & \vdots \\ h1n & \cdots & hnn \end{bmatrix} *\mathrm{inv}\Big(\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix} * \begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k * \begin{bmatrix} h11 & \cdots & hn1 \\ \vdots & \ddots & \vdots \\ h1n & \cdots & hnn \end{bmatrix} + R \Big)$$

这里设

$$Kgd = \begin{bmatrix} Kgd11 & \cdots & Kgd1n \\ \vdots & \ddots & \vdots \\ Kgdn1 & \cdots & Kgdnn \end{bmatrix} = \begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix} * \begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k * \begin{bmatrix} h11 & \cdots & hn1 \\ \vdots & \ddots & \vdots \\ h1n & \cdots & hnn \end{bmatrix} + R$$

[0110] 其中 $Kgd11=((h11*P_{pre11(k)}+\cdots+h1n*P_{pre1n(k)})*h11+\cdots+(h11*P_{pre1n(k)}+\cdots+h1n*P_{prenn(k)})*h1n)+R,\cdots,Kgd1n=((h11*P_{pre11(k)}+\cdots+h1n*P_{pre1n(k)})*hn1+\cdots+(h11*P_{pre1n(k)}+\cdots+h1n*P_{prenn(k)})*hnn)+R,\cdots,Kgdn1=((hn1*P_{_pre11(k)}+\cdots+hnn*P_{_pren1(k)})*h11+\cdots+(hn1*P_{pre1n(k)}+\cdots+hnn*P_{prennk})*h1n)+R,\cdots,Kgdnn=((hn1*P_{_pre11(k)}+\cdots+hnn*P_{_pren1(k)})*hn+,\cdots+(hn1*P_{_pre1n(k)}+\cdots+hnn*P_{_prenn(k)})*hnn)+R$ 。则

$$\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix} = \begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k * \begin{bmatrix} h11 & \cdots & hn1 \\ \vdots & \ddots & \vdots \\ h1n & \cdots & hnn \end{bmatrix} *\mathrm{inv}\Big(\begin{bmatrix} Kgd11 & \cdots & Kgd1n \\ \vdots & \ddots & \vdots \\ Kgdn1 & \cdots & Kgdnn \end{bmatrix} \Big) =$$

$$\begin{bmatrix}(P_pre11(k)*h11+\cdots+P_pre1n(k)*h1n) & \cdots & (P_pre11(k)*hn1+\cdots+P_pre1n(k)*hnn)\\ \vdots & \ddots & \vdots\\ (P_pren1(k)*h11+\cdots+P_prenn(k)*h1n) & \cdots & (P_pren1(k)*hn1+\cdots+P_prenn(k)*hnn)\end{bmatrix}*$$
$$inv(\begin{bmatrix}Kgd11 & \cdots & Kgd1n\\ \vdots & \ddots & \vdots\\ Kgdn1 & \cdots & Kgdnn\end{bmatrix})$$
。

[0111] （H）状态最优估算值计算部分用来计算产生状态最优估算值 Xkf(k) $=\begin{bmatrix}Xkf1\\ \vdots\\ Xkfn\end{bmatrix}_k$，状态最优估算值是由（F）中计算出的状态预测值 X_pre(k) $=\begin{bmatrix}X_pre1\\ \vdots\\ X_pren\end{bmatrix}_k$ 和(G)中计算出的的卡尔曼增益 Kg $=\begin{bmatrix}Kg11 & \cdots & Kg1n\\ \vdots & \ddots & \vdots\\ Kgn1 & \cdots & Kgnn\end{bmatrix}$ 和(D)计算出的观察测量值 Z(K) $=\begin{bmatrix}z1\\ \vdots\\ zn\end{bmatrix}_k$ 和(B) 计算出的观察矩阵 H= $\begin{bmatrix}h11 & \cdots & h1n\\ \vdots & \ddots & \vdots\\ hn1 & \cdots & hnn\end{bmatrix}$ 计算得出。其计算方程为 Xkf(k)＝X_pre(k)+Kg(k)＊(Z(k)-H*X_pre(k))。

[0112] 表示成矩阵形式为，

$$\begin{bmatrix}Xkf1\\ \vdots\\ Xkfn\end{bmatrix}_k=\begin{bmatrix}X_pre1\\ \vdots\\ X_pren\end{bmatrix}_k+\begin{bmatrix}Kg11 & \cdots & Kg1n\\ \vdots & \ddots & \vdots\\ Kgn1 & \cdots & Kgnn\end{bmatrix}_k *(\begin{bmatrix}z1\\ \vdots\\ zn\end{bmatrix}_k-\begin{bmatrix}h11 & \cdots & h1n\\ \vdots & \ddots & \vdots\\ hn1 & \cdots & hnn\end{bmatrix}*\begin{bmatrix}X_pre1\\ \vdots\\ X_pren\end{bmatrix}_k)=$$

[0113]

$$\begin{bmatrix}X_pre1\\ \vdots\\ X_pren\end{bmatrix}_k+\begin{bmatrix}Kg11 & \cdots & Kg1n\\ \vdots & \ddots & \vdots\\ Kgn1 & \cdots & Kgnn\end{bmatrix}_k*\begin{bmatrix}z1(k)-(h11*X_{_pre1}(k)+\cdots+h1n*X_pren(k))\\ \vdots\\ zn(k)-(hn1*X_{_pre1(k)}+\cdots+hnn*X_pren(k))\end{bmatrix}$$
$$=$$
$$\begin{bmatrix}X_{pre1(k)}+(Kg11*(z1(k)-(h11*X_{pre1}(k)+\cdots+h1n*X_pren(k)))+\cdots+Kg1n*(zn(k)-(h1n*X_{_pre1(k)}+\cdots+hnn*X_pren(k)))\\ \vdots\\ X_{pren(k)}+(Kgn1*(z1(k)-(h11*X_{pre1}(k)+\cdots+h1n*X_pren(k)))+\cdots+Kgnn*(zn(k)-(h1n*X_{_pre1(k)}+\cdots+hnn*X_pren(k)))\end{bmatrix}$$

[0114] (I)状态协方差更新部分是通过(G)生成的卡尔曼增益 Kg $=\begin{bmatrix}Kg11 & \cdots & Kg1n\\ \vdots & \ddots & \vdots\\ Kgn1 & \cdots & Kgnn\end{bmatrix}$ 和(F)生成的状态预测协方差 P_pre(k)=$\begin{bmatrix}P_{_pre11} & \cdots & P_{_pre1n}\\ \vdots & \ddots & \vdots\\ P_{_pren1} & \cdots & P_{_prenn}\end{bmatrix}_k$ 和(B)计算出观察矩阵

$H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$通过运 算来更新状态协方差 $P(k)=\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_k$ 。其计算方程为 P(k)＝(I-Kg(k)*H)*P_pre(k)。表示成矩阵形式为，

$$\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_k = \left(\begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix} - \begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_k * \begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}\right)*$$

[0115] $$\begin{bmatrix} P_{_pre11} & \cdots & P_{_pre1n} \\ \vdots & \ddots & \vdots \\ P_{_pren1} & \cdots & P_{_prenn} \end{bmatrix}_k = \begin{bmatrix} 1-(Kg11*h11+\cdots+Kg1n*hn1) & \cdots & -(Kg11*h1n+\cdots+Kg1n*hnn) \\ \vdots & \ddots & \vdots \\ -(Kgn1*h11+\cdots+Kgnn*hn1) & \cdots & 1-(Kgn1*h1n+\cdots+Kgnn*hnn) \end{bmatrix}$$

$$*\begin{bmatrix} P_{pre11} & \cdots & P_{_pre1n} \\ \vdots & \ddots & \vdots \\ P_{_pren1} & \cdots & P_{_prenn} \end{bmatrix}_k。$$

[0116] 其中 P11(k)＝((1-(Kg11(k)*h11+…+Kg1n(k)*hn1))*P_pre11(k)+…+ (-(Kg11(k)*h1n+…+Kg1n(k)*hnn))*P_{-pren1}(k)),…,P1n＝((1-(Kg11(k)*h11+ …+Kg1n(k)*hn1)*P_{-pre1n}(k)+…+(-(Kg11(k)*h1n+…+Kg1n(k)*hnn))* P_{-prenn}(k))),…,Pn1(k)＝((-(Kgn1(k)*h11+…+Kgnn(k)*hn1))*$P_{_pre11}$(k)+ …+(1-(Kgn1(k)*h1n+…+Kgnn(k)*hnn))*P_{-pren1}(k)),…,Pnn＝((-(Kgn1(k)* h11+…+Kgnn(k)*hn1))*P_{-pre1n}(k)+…+(1-(Kgn1(k)*h1n+…+Kgnn(k)* hnn))*P_{-prenn}(k)))

[0117] 。

[0118] (J)(K)分别为是为了产生测量过程中遇到的噪声和状态转移过程中产 生的噪声 $V(K)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_k$ 和 $W(K)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_k$，这两个噪声的协方差分别为 (M)(L)产生的协方差R和 $Q=\begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix}$。$V(K)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_k$，$W(K)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_k$，R，$Q=\begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix}$根据测量的实际情况和经验公式计算产生，公式的具体形式因系统和测量过程的不同而不同。

[0120] 本发明实施例公开一种并行卡尔曼滤波系统，参照图1～5所示，包括传输 矩阵生成模块、观察矩阵生成模块、状态转移计算模块、观察测量计算模块、 状态预测生成模块、预测协方差生成模块、卡尔曼增益计算模块、状态最优估 算值计算模块、状态协方差更新模块、测量过程中的噪声生成模块、状态转移 过程中产生的噪声生成模块、过程噪声协方差生成模块和测量噪声协方差生成 模块。状态转移计算模块、观察测量计算模块、状态预测生成模块、预测协方 差生成模块和卡尔曼增益计算模块均与传输矩阵生成模块和观察矩阵生成模块 连接。状态最优估算值计算模块和状态协方差更新模块均与观察矩阵生成模块 连接。预测协方差生成模块、卡尔曼增益计算模块、测量过程中的噪声生成模 块和状态转移过程中产生的噪声生成模块均与过程噪声协方差生成模块和测量 噪声协方差生成

模块连接。测量过程中的噪声生成模块和状态转移过程中产生 的噪声生成模块分别与观察测量计算模块和状态转移计算模块连接。

[0121] (1)传输矩阵生成模块(AModule)能够完成步骤(A)中功能。

[0122] (2)观察矩阵生成模块(HModule)能够完成步骤(B)中功能。

[0123] (3)将状态转移计算模块(StateEquationModule)用并行的多个独立的子 模块(SESubModule)来实现:状态转移计算子模块1(SESubModule1),计算 X(1):X(1)＝系统初始状态。状态转移计算子模块2(SESubModule2),计算 X(2):X(2)＝A*X(1)+L*W(1)。状态转移计算子模块N-1(SESubModuleN-1),计 算X(N-1):X(N-1)＝A*X(N-2)+L*W(N-2)。状态转移计算子模块N (SESubModuleN),计算X(N):X(N)＝A*X(N-1)+L*W(N-1)。每一个状态转移计 算子模块都能够完成步骤(C)中所述的完整功能X(k)＝A*X(k-1)+L*W(k-1)，这样N个状态转移计算子模块就可以同时计算出N个不同的系统状态X(N)。当N<K时,用N个子模块计算完N个不同的系统状态X(1),X(2)…X(N-1)，X(N)之后,就开始新一轮的计算,再计算出N个不同的系统状态X(N+1)，X(N+2)…X(2N-1),X(2N),如此往复直到系统的K个状态X(1),X(2)…X(K-1)，X(K)全部被计算出来。当N>＝K时,可以用N个子模块中的K个子模块直接 一次将系统的K个状态X(1),X(2)…X(K-1)X(K)全部计算出来。

[0124] (4)将观察测量计算模块(ObserveEquationModule)用并行的多个独立的 子模块(OESubModule)来实现:观察测量计算子模块1(OESubModule1)，计算Z(1):Z(1)＝H*X(1)+V(1)。观察测量计算子模块2(OESubModule2),计算 Z(2):Z(2)＝H*X(2)+V(2)。观察测量计算子模块N-1(OESubModuleN-1),计算 Z(N-1):Z(N-1)＝H*X(N-1)+V(N-1)。观察测量计算子模块N(SESubModuleN)，计算Z(N):Z(N)＝H*X(N)+V(N)。每一个观察测量计算子模块都能够完成步骤 (D)中所述的完整功能Z(k)＝H*X(k)+V(k),这样N个观察测量计算子模块就可以同时计算出N个不同的观察测量值Z(N)。当N<K时,用N个子模块计算 完N个不同的观察测量值Z(1),Z(2)…Z(N-1),Z(N)之后,就开始新一轮的计 算,再计算出N个不同观察测量值Z(N+1),Z(N+2)…Z(2N-1),Z(2N),如此往 复直到系统的K个观察测量值X(1),X(2)…X(K-1),X(K)全部被计算出来。当 N>＝K时,可以用N个子模块中的K个子模块直接一次将系统的K个观察测量 值Z(1),Z(2)…Z(K-1)Z(K)全部计算出来。

[0125] (5)将状态预测生成模块(StatePredicteEquationModule)用并行的多个独 立的子模块(SPESubModule)来实现:状态预测生成子模块1(SPESubModule1)，计算X_pre(1):X_pre(1)＝系统状态预测初始值。状态预测生成子模块2 (SPESubModule2),计算X_pre(2):X_pre(2)＝A*Xkf(1)。状态预测生成子模块 N-1(SPESubModuleN-1),计算X_pre(N-1):X_pre(N-1)＝A*Xkf(N-2)。状态预测 生成子模块N(SPESubModuleN),计算X_pre(N):X_pre(N)＝A*Xkf(N-1)。每一 个状态预测生成子模块都能够完成步骤(E)中所述的完整功能X_pre(k)＝A*Xkf(k-1),这样N个状态预测生成子模块就可以同时计算出N个不 同的状态预测值X_pre(N)。当N<K时,用N个子模块计算完N个不同的状态 预测值X_pre(1),X_pre(2)…X_pre(N-1),X_pre(N)之后,就开始新一轮的计算，再计算出N个不同状态预测值X_pre(N+1),X_pre(N+2)…X_pre(2N-1)，X_pre(2N),如此往复直到系统的K个状态预测值X_pre(1),X_pre(2)… X_pre(K-1),X_pre(K)全部被计算出来。当N>＝K时,可以用N个子模块中的 K个子模块直接一次将系统的K个状态预测值X_pre(1),X_pre(2)… X_pre(K-1)X_pre(K)全部计算出来。

[0126] (6)将预测协方差生成模块(ConvariancePredicteEquationModule)用并行 的多个独立的子模块(CPESubModule)来实现:预测协方差生成子模块1 (SPESubModule1),计算P_pre(1):P_pre(1)=系统预测协方差初始值。预测协方 差生成子模块2(SPESubModule2),计算P_pre(2):P_pre(2)=A*P(1)*A'+Q。预 测协方差生成子模块N-1(SPESubModuleN-1),计算P_pre(N-1): P_pre(N-1)=A*P(N-2)*A'+Q。预测协方差生成子模块N(SPESubModuleN),计 算P_pre(N):P_pre(N)=A*P(N-1)*A'+Q。每一个预测协方差生成子模块都能够 完成步骤(F)中所述的完整功能P_pre(k)=A*P(k-1)*A'+Q,这样N个预测协方 差生成子模块就可以同时计算出N个不同的预测协方差值P_pre(N)。当N<K 时,用N个子模块计算完N个不同的预测协方差值P_pre(1),P_pre(2)… P_pre(N-1),P_pre(N)之后,就开始新一轮的计算,再计算出N个不同预测协方 差值P_pre(N+1),P_pre(N+2)…P_pre(2N-1),P_pre(2N),如此往复直到系统的 K个预测协方差值P_pre(1),P_pre(2)…P_pre(K-1),P_pre(K)全部被计算出来。当N>=K时,可以用N个子模块中的K个子模块直接一次将系统的K个预测 协方差值P_pre(1),P_pre(2)…P_pre(K-1),P_pre(K)全部计算出来。

[0127] (7)将卡尔曼增益计算模块(KalmanGainEquationModule)用并行的多个 独立的子模块(KGESubModule)来实现:卡尔曼增益计算子模块1 (KGESubModule1),计算Kg(1):Kg(1)=卡尔曼增益的初始值。卡尔曼增益计算 子模块2(KGESubModule2),计算Kg(2):Kg(2)=P_pre(2)*H'*inv(H*P_pre(2)*H'+R)。卡尔曼增益计算子模块N-1 (KGESubModuleN-1),计算Kg(N-1): Kg(N-1)=P_pre(N-1))*H'*inv(H*P_pre(N-1))*H'+R)。卡尔曼增益计算子模块N (KGESubModuleN),计算Kg(N): Kg(N)=P_pre(N))*H'*inv(H*P_pre(N))*H'+R)。每一个卡尔曼增益计算子模块都 能够完成步骤(G)中所述的完整功能Kg(k)=P_pre(k)*H'*inv(H*P_pre(k)*H'+R), 这样N个卡尔曼增益计算子模块就可以同时计算出N个不同的卡尔曼增益值 Kg(N)。当N<K时,用N个子模块计算完N个不同的卡尔曼增益值Kg(1), Kg(2)… Kg(N-1),Kg(N)之后,就开始新一轮的计算,再计算出N个不同卡尔 曼增益值Kg(N+1),Kg(N+2)…Kg(2N-1),Kg(2N),如此往复直到系统的K个 卡尔曼增益值Kg(1),Kg(2)…Kg(K-1),Kg(K)全部被计算出来。当N>=K时, 可以用N个子模块中的K个子模块直接一次将系统的K个卡尔曼增益值Kg(1), Kg(2)…Kg(K-1),Kg(K)全部计算出来。

[0128] (8)将状态最优估算值计算模块(StateUpdateEquationModule)用并行的 多个独立的子模块(SUESubModule)来实现:状态最优估算值计算子模块1 (KGESubModule1),计算Xkf(1):Xkf(1)=状态最优化估算值初始值。状态最优 估算值计算子模块2(KGESubModule2),计算Xkf(2): Xkf(2)=X_pre(2)+Kg(2)*(Z(2)-H*X_pre(2)。状态最优估算值计算子模块N-1 (KGESubModuleN-1),计算Xkf(N-1): Xkf(N-1)=X_pre(N-1)+Kg(N-1)*(Z(N-1)-H*X_pre(N-1)。状态最优估算值计算子 模块N(KGESubModuleN),计算Xkf(N): Xkf(N)=X_pre(N)+Kg(N)*(Z(N)-H*X_pre(N)。每一个状态最优估算值计算子模 块都能够完成步骤(H)中所述的完整功能 Xkf(k)=X_pre(k)+Kg(k)*(Z(k)-H*X_pre(k),这样N个状态最优估算值计算子模 块就可以同时计算出N个不同的状态最优估算值Xkf(N)。当N<K时,用N个 子模块计算完N个不同的状态最优估算值Xkf(1),Xkf(2)…Xkf(N-1),Xkf(N) 之后,就开始新一轮的计算,再计算出N个不同状态最优估算值Xkf(N+1), Xkf(N+2)… Xkf(2N-1),Xkf(2N),如此往复直到系统的K个状态最优估算值 Xkf(1),Xkf(2)…Xkf(K-1),Xkf(K)全部被计算出来。当N>=K时,可以用N个 子模块中的K个子模块直接一次将系统

的K个状态最优估算值Xkf(1),Xkf(2)… Xkf(K-1),Xkf(K)全部计算出来。

[0129] (9)将状态协方差更新模块(ConvarianceUpdateEquationModule)用并行 的多个独立的子模块(CUESubModule)来实现:状态协方差更新子模块1 (CUESubModule1),计算P(1):P(1)=状态协方差初始值。状态协方差更新子模 块2(CUESubModule2),计算P(2):P(2)=(I-Kg(2)*H)*P_pre(2)。状态协方差更 新子模块N-1(CUESubModuleN-1),计算 P(N-1):P(N-1)=(I-Kg(N-1)*H)*P_pre(N-1)。状态协方差更新子模块N (CUESubModuleN),计算P(N):P(N)=(I-Kg(N)*H)*P_pre(N)。每一个状态协方 差更新子模块都能够完成步骤(I)中所述的完整功能 P(k)=(I-Kg(k)*H)*P_pre(k),这样N个状态协方差更新子模块就可以同时计算出 N个不同的状态协方差更新值P(N)。当N<K时,用N个子模块计算完N个不同的状态协方差更新值P(1),P(2)…P(N-1),P(N)之后,就开始新一轮的计算, 再计算出N个不同状态协方差更新值P(N+1),P(N+2)…P(2N-1),P(2N),如此 往复直到系统的K个状态协方差更新值P(1),P(2)…P(K-1),P(K)全部被计算 出来。当N>=K时,可以用N个子模块中的K个子模块直接一次将系统的K 个状态协方差更新值P(1),P(2)…P(K-1),P(K)全部计算出来。

[0130] (10)将测量过程中的噪声生成模块(VGenerateModule)用并行的多个独 立的子模块(VGSubModule)来实现:测量过程中噪声生成子模块1 (VGSubModule1),计算V(1)。测量过程中噪声生成子模块2(VGSubModule2), 计算V(2)。测量过程中噪声生成子模块N-1(VGSubModuleN-1),计算V(N-1)。 测量过程中噪声生成子模块N(VGSubModuleN),计算V(N)。每一个测量过 程中噪声生成子模块都能够完成步骤(J)中所述的完整功能,这样N个测量 过程中噪声生成子模块就可以同时计算出N个不同的测量过程噪声值V(N)。当 N<K时,用N个子模块计算完N个不同的测量过程噪声值V(1),V(2)…V(N-1), V(N)之后,就开始新一轮的计算,再计算出N个不同测量过程噪声值V(N+1), V(N+2)…V(2N-1),V(2N),如此往复直到系统的K个测量过程噪声值V(1), V(2)…V(K-1),V(K)全部被计算出来。当N>=K时,可以用N个子模块中的K 个子模块直接一次将系统的K个测量过程噪声值V(1),V(2)…V(K-1),V(K) 全部计算出来。

[0131] (11)将状态转移过程中产生的噪声生成模块(WGenerateModule)用并 行的多个独立的子模块(WGSubModule)来实现:传输噪声生成子模块1 (WGSubModule1),计算W(1)。传输噪声生成子模块2(WGSubModule2), 计算W(2)。传输噪声生成子模块N-1(WGSubModuleN-1),计算W(N-1)。传 输噪声生成子模块N(WGSubModuleN),计算W(N)。每一个传输噪声生成子 模块都能够完成步骤(K)中所述的完整功能,这样N个传输噪声生成子模块 就可以同时计算出N个不同的传输噪声值W(N)。当N<K时,用N个子模块计 算完N个不同的传输噪声值W(1),W(2)…W(N-1),W(N)之后,就开始新一轮 的计算,再计算出N个不同传输噪声值W(N+1),W(N+2)…W(2N-1),W(2N), 如此往复直到系统的K个传输噪声值W(1),W(2)…W(K-1),W(K)全部被计算 出来。当N>=K时,可以用N个子模块中的K个子模块直接一次将系统的K 个传输噪声值W(1),W(2)…W(K-1),W(K)全部计算出来。

[0132] (12)过程噪声协方差生成模块(QGenerateModule)能够完成步骤(L) 中功能。

[0133] (13)测量噪声协方差生成模块(RGenerateModule)能够完成步骤(M) 中功能。

[0134] 具体而言,参照图5所示,首先将图中各模块名称说明如下:

[0135] AM(AModule):传输矩阵生成模块。HM(HModule):观察矩阵生成模 块。SESM

(StateEquationSubModule)状态转移计算子模块。OESM (ObserveEquationSubModule)：观察测量计算子模块。SPESM (StatePredicteEquationSubModule)：状态预测生成子模块。CPESM (ConvariancePredicteEquationSubModule)：预测协方差生成子模块。KGESM (KalmanGainEquationSubModule)：卡尔曼增益计算子模块。SUESM (StateUpdateEquationSubModule)：状态最优估算值计算子模块。CUESM (ConvarianceUpdateEquationSubModule)：状态协方差更新子模块。VGSM (VGenerateSubModule)：测量噪声生成子模块。WGSM (WGenerateSubModule)：传输噪声生成子模块。RGM (RGenerateModule)：过程噪声协方差生成模块。QGM(QGenerateModule)：测量噪声协方差生成模 块。

[0136] 针对于上述的各模块或子模块的说明均是就其所完成的功能而言，至于上 述的各模块或子模块实现：如果是ASIC，上述的各模块或子模块所对应就是专 用集成电路相应的功能模块；如果是多核CPU/GPU，上述的各模块或子模块所 对应的就是完成该功能所使用的一个或多个CPU/GPU核；如果是FPGA，上述 的各模块或子模块所对应的就是实现该功能的一个或多个FPGA单元(如 DSPblock、LUT、BRAM等)。

[0137] 并行卡尔曼滤波子模块间数据传递及运算方法，包括以下步骤：

[0138] 1)AM(AModule-传输矩阵生成模块)计算生成的传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$通过路径1_1，1_2，…，1_n-1，1_n分别传送给SESM (StateEquationSubModule-状态转移计算子模块)1，SESM2，…，SESMn-1， SESMn。

[0139] 2)AM计算生成的传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$通过路径2_1，2_2，…，2_n-1， 2_n分别传送给SPESM(StatePredicteEquationSubModule-状态预测生成子模块) 1，SPESM2，…，SPESMn-1，SPESMn。

[0140] 3)AM计算生成的传输矩阵 $A=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}$通过路径3_1，3_2，…，3_n-1， 3_n分别传送给CPESM(ConvariancePredicteEquationSubModule-预测协方差生 成子模块)1，CPESM2，…，CPESMn-1，CPESMn。

[0141] 4)HM(HModule-观察矩阵生成模块)计算生成的观察矩阵 $H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$通过路径4_1，4_2，…，4_n-1，4_n分别传送给OESM (ObserveEquationSubModule-观察测量计算子模块)1，OESM2，…，OESMn-1， OESMn。

[0142] 5)HM计算生成的观察矩阵 $H=\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix}$通过路径5_1，5_2，…，5_n-1， 5_n分别传送给KGESM(KalmanGainEquationSubModule-卡尔曼增益计算子模 块)1，KGESM2，…，KGESMn-1，KGESMn。

[0143] 6)SESM1计算生成的X(1)＝系统初始状态通过路径6_1传送给SESM2，SESM2计算生成的 $X(2)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_2$ 通过路径6_2传送给SESM3,SESMn-1计算生 成的 $X(n-1)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_{n-1}$ 通过路径6_n-1传送给SESMn,SESMn计算生 $X(n)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_n$ 通 过路径6_n传送给SESM1(如图3(3)中所述,当N<K时,进行此操作。当 N>＝K时,SESMk计算生成 $X(k)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_k$ 便结束操作)。

[0144] 7)WGSM(QGenerateModule-测量噪声协方差生成模块)1计算生成的 $W(1)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_1$ 通过路径7_1传送给SESM2,WGSM2计算生 $W(2)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_2$ 通过路径 7_2传送给SESM3,WGSMn-1计算生 $W(n-1)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_{n-1}$ 通过路径7_n-1传送 给SESMn,WGSMn计算生 $W(n)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_n$ 通过路径7_n传送给SESM1(当 N<K时,进行此操作。当N>＝K时,WGSMk计算生 $W(k)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_k$ 便结束操 作)。

[0145] 8)SESM1计算生 $X(1)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_1$ 通过路径8_1传送给OESM1,SESM2计算生 $X(2)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_2$ 通过路径8_2传送给OESM2,SESMn-1计算生 $X(n-1)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_{n-1}$ 通过路 径8_n-1传送给OESMn-1,SESMn计算生 $X(n)=\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_n$ 通过路径8_n传送给 OESMn。

[0146] 9)VGSM(VGenerateSubModule-测量噪声生成子模块)1计算生成的 $V(1)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_1$ 通过路径9_1传送给OESM1,VGSM2计算生成的 $V(2)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_2$ 通过路 径9_2传送给OESM2,VGSMn-1计算生成的 $V(n-1)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_{n-1}$ 通过路径9_n-1 传送给OESMn-1,VGSMn计算生成

的 $V(n)=\begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_n$ 通过路径9_n传送给 OESMn。

[0147] 10)SUESM(StateUpdateEquationSubModule-状态最优估算值计算子模块) 1计算生成的 $Xkf(1)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_1$ 通过路径10_1传送给 SPESM (StatePredicteEquationSubModule-状态预测生成子模块)2,SUESM2计算 生成的 $Xkf(2)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_2$ 通过路径10_2传送给SPESM3,SUESMn-1计算生成的 $Xkf(n-1)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_{n-1}$ 通过路径10_n-1传送给SPESMn,SUESMn计算生成的 $Xkf(n)=\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_n$ 通过路径10_n传送给SPESM1(当N<K时,进行此操作。当 N>=K时,SUESMk 计算生 $W(k)=\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_k$ 便结束操作)。

[0148] 11)CUESM(ConvarianceUpdateEquationSubModule-状态协方差更新子模 块)1计算生成的 $P(1)=\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_1$ 通过路径11_1传送给CPESM2,CUESM2 计算生 $P(2)=\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_2$ 通过路径11_2传送给CPESM3,CUESMn-1计算生 成的 $P(n-1)=\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_{n-1}$ 通过路径11_n-1传送给CPESMn,CUESMn计算 生成的 $P(n)=\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_n$ 通过路径11_n传送给CPESM1(当N<K时,进行此 操作。当N>=K时, CUESMk计算生 $P(k)=\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_k$ 便结束操作)。

[0149] 12)QGM(QGenerateModule-测量噪声协方差生成模块)计算生成的过程 噪声协方差矩阵 $Q=\begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix}$ 通过路径12_1,12_2,…,12_n-1,12_n分 别传送给CPESM1,

CPESM2,…,CPESMn-1,CPESMn。

[0150] 13)CPESM1计算生成的P_pre(1)＝系统预测协方差初始值通过路径13_1 传送给KGESM1,CPESM2计算生成的 $P_pre(2)=\begin{bmatrix} P_{_pre11} & \cdots & P_{_pre1n} \\ \vdots & \ddots & \vdots \\ P_{_pren1} & \cdots & P_{_prenn} \end{bmatrix}_2$ 通过路径 13_2传送给KGESM2,CPESMn-1计算生成的 $P_pre(n-1)=\begin{bmatrix} P_{_pre11} & \cdots & P_{_pre1n} \\ \vdots & \ddots & \vdots \\ P_{_pren1} & \cdots & P_{_prenn} \end{bmatrix}_{n-1}$ 通过路径13_n-1传送给KGESMn-1,CPESMn计算生成的 $P_pre(n)=\begin{bmatrix} P_{_pre11} & \cdots & P_{_pre1n} \\ \vdots & \ddots & \vdots \\ P_{_pren1} & \cdots & P_{_prenn} \end{bmatrix}_n$ 通过路径13_n传送给KGESMn。

[0151] 14)RGM(RGenerateModule-过程噪声协方差生成模块)计算生成的测量协 方差R通过路径14_1,14_2,…,14_n-1,14_n分别传送给KGESM1, KGESM2,…,KGESMn-1,KGESMn。

[0152] 15)SPESM1计算生成的 $X_pre(1)=\begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_1$ 通过路径15_1传送给 SUESM1,SPESM2计算生成的 $X_pre(2)=\begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_2$ 通过路径15_2传送给 SUESM2,SPESMn-1计算生成的 $X_pre(n-1)=\begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_{n-1}$ 通过路径15_n-1传送 给SUESMn-1,SPESMn计算生成的 $X_pre(n)=\begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_n$ 通过路径15_n传送给 SUESMn。

[0153] 16)KGESM1计算生成的 $Kg(1)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_1$ 通过路径16_1传送给SUESM1,KGESM2计算生成的 $Kg(2)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_2$ 通过路径16_2传送 给SUESM2,KGESMn-1计算生成的 $Kg(n-1)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_{n-1}$ 通过路径 16_n-1传送给SUESMn-1,KGESMn计算生成的 $Kg(n)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_n$ 通 过路径16_n传送给SUESMn。

[0154] 17)OESM1计算生成的 $Z(1)=\begin{bmatrix} z1 \\ \vdots \\ zn \end{bmatrix}_1$ 通过路径17_1传送给SUESM1,OESM2 计算生成的 $Z(2)=\begin{bmatrix} z1 \\ \vdots \\ zn \end{bmatrix}_2$ 通过路径17_2传送给SUESM2,OESMn-1计算生成的 $Z(n-1)=\begin{bmatrix} z1 \\ \vdots \\ zn \end{bmatrix}_{n-1}$ 通过路径17_n-1传送给SUESMn-1,OESMn计算生成的 $Z(n)=\begin{bmatrix} z1 \\ \vdots \\ zn \end{bmatrix}_n$ 通过路径17_n传送给SUESMn。

[0155] 18)KGESM1计算生成的 $Kg(1)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_1$ 通过路径18_1传送给CUESM1,KGESM2计算生成的 $Kg(2)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_2$ 通过路径18_2传送 给CUESM2,KGESMn-1计算生成的Kg(n-1) $Kg(n-1)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_{n-1}$ 通过路径18_n-1传送给CUESMn-1,KGESMn计算生成的 $Kg(n)=\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_n$ 通过路径18_n传送给CUESMn。

[0156] 19)CPESM1计算生成的 $P_pre(1)=\begin{bmatrix} P_{_pre11} & \cdots & P_{_pre1n} \\ \vdots & \ddots & \vdots \\ P_{_pren1} & \cdots & P_{_prenn} \end{bmatrix}_1$ 通过路径19_1传送 给CUESM1,CPESM2计算生成的 $P_pre(2)=\begin{bmatrix} P_{_pre11} & \cdots & P_{_pre1n} \\ \vdots & \ddots & \vdots \\ P_{_pren1} & \cdots & P_{_prenn} \end{bmatrix}_2$ 通过路径19_2 传送给CUESM2,CPESMn-1计算生成的 $P_pre(n-1)=\begin{bmatrix} P_{_pre11} & \cdots & P_{_pre1n} \\ \vdots & \ddots & \vdots \\ P_{_pren1} & \cdots & P_{_prenn} \end{bmatrix}_{n-1}$ 通过 路径19_n-1传送给CUESMn-1,CPESMn计算生成的 $P_pre(n)=\begin{bmatrix} P_{_pre11} & \cdots & P_{_pre1n} \\ \vdots & \ddots & \vdots \\ P_{_pren1} & \cdots & P_{_prenn} \end{bmatrix}_n$ 通过路径19_n传送给CUESMn。

[0157] 20)RGM计算生成的测量噪声协方差R通过路径20_1,20_2,…,20_n-1, 20_n分别传送给VGSM1,VGSM2,…,VGSMn-1,VGSMn。

[0158] 21)QGM计算生成的过程噪声协方差矩阵 $Q=\begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix}$ 通过路径21_1，21_2,…,21_n-1,21_n分别传送给WGSM1,WGSM2,…,WGSMn-1,WGSMn。

[0159] 在本发明一个优选的实施例中,用并行卡尔曼滤波实现20000个状态点的 载波追踪。选用多普勒频偏DetaTheta,频偏的变化率Omega0,频偏变化率的 变化率Omega1作为系统的状态量,所以我们举例的系统为三阶系统,我们取 detav=1e-3,detaw=1e-3,Ts(采样时间)=1e-3, $L(\text{噪声控制矩阵})=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$。观察矩阵生成模块(HM)计算观测矩阵的经验公式为 H=[1 Ts/2 Ts^2/6]=[1 5.0e-04 1.67e-07]。传输矩阵生成模块(AM)计算观测矩阵的经验公式为 $A=\begin{bmatrix} 1 & Ts & Ts^2/2 \\ 0 & 1 & Ts \\ 0 & 0 & 1 \end{bmatrix}=\begin{bmatrix} 1 & 1.0e-03 & 5.0e-07 \\ 0 & 1 & 1.0e-03 \\ 0 & 0 & 1 \end{bmatrix}$。测量噪声协方差生成模块(QGM)计算过程噪声协方差矩阵的经验公式为 $Q=detav^2*\begin{bmatrix} Ts^6/252 & Ts^5/72 & Ts^4/30 \\ Ts^5/72 & Ts^4/20 & Ts^3/8 \\ Ts^4/30 & Ts^3/8 & Ts^2/3 \end{bmatrix}=\begin{bmatrix} 3.9689e-27 & 1.39e-23 & 3.33e-20 \\ 1.39e-23 & 5.00e-20 & 1.25e-16 \\ 3.33e-20 & 1.25e-16 & 3.33e-13 \end{bmatrix}$。

[0160] 过程噪声协方差生成模块(RGM)计 算测量噪声协方差的经验公式为R=detaw^2/Ts^4=1.0e+06。测量噪声生成子模 块(VGSM)计算测量噪声V=N(0,R)。传输噪声生成子模块(WGSM)计 算传输噪声W=N(0,Q)。设定系统状态量的初始值为X(1)=[0 -20*e3 0]，系统观测量的初始值为Z(1)=[0 0 0]，系统状态协方差初始值 $P(1)=\begin{bmatrix} 1/12 & 0 & 0 \\ 0 & 1*10^{(-4)} & 0 \\ 0 & 0 & 1*10^{(-4)} \end{bmatrix}$，预测协方差初始值 $P_pre(1)=\begin{bmatrix} 1/12 & 0 & 0 \\ 0 & 1*10^{(-4)} & 0 \\ 0 & 0 & 1*10^{(-4)} \end{bmatrix}$，状态最优估算值的初始值Xkf (1)=[0 -20*10^(3) 0],状态预测值的初始值X_pre(1)=[0 0 0]。参照图6所示,取n=500,即将卡尔曼滤波的每个步骤用500个子模块进行并 行运算。

[0161] 如上所述,在本示例中,SESMk完成X(k)=A*X(k-1)+L*W(k-1),表示成 矩阵形式为，$\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_k=\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}*\begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_{k-1}+\begin{bmatrix} l11 & \cdots & l1n \\ \vdots & \ddots & \vdots \\ ln1 & \cdots & lnn \end{bmatrix}*\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_{k-1} \Rightarrow \begin{bmatrix} x1 \\ \vdots \\ x3 \end{bmatrix}_k=\begin{bmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{bmatrix}*\begin{bmatrix} x1 \\ \vdots \\ x3 \end{bmatrix}_{k-1}+\begin{bmatrix} l11 & l12 & l13 \\ l21 & l22 & l23 \\ l31 & l32 & l33 \end{bmatrix}*\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_{k-1} \Rightarrow \begin{bmatrix} x1 \\ \vdots \\ x3 \end{bmatrix}_k=\begin{bmatrix} 1 & Ts & Ts^2/2 \\ 0 & 1 & Ts \\ 0 & 0 & 1 \end{bmatrix}*\begin{bmatrix} x1 \\ \vdots \\ x3 \end{bmatrix}_{k-1}+\begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}*\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_{k-1} \Rightarrow$

[0162] $\begin{bmatrix} x1 \\ \vdots \\ x3 \end{bmatrix}_k=\begin{bmatrix} 1 & 1.0e-03 & 5.0e-07 \\ 0 & 1 & 1.0e-03 \\ 0 & 0 & 1 \end{bmatrix}*\begin{bmatrix} x1 \\ \vdots \\ x3 \end{bmatrix}_{k-1}+\begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}*\begin{bmatrix} w1 \\ \vdots \\ wn \end{bmatrix}_{k-1}$，表示成方程形式为，

[0163] x1(k)=a11*x1(k-1)+a12*x2(k-1)+a13*x3(k-1)+l11*w1(k-1)+l12*w2(k-1)+

113* w3(k-1)，

[0164] x2(k)＝a21*x1(k-1)+a22*x2(k-1)+a23*x3(k-1)+121w1(k-1)+122*w2(k-1)+123* w3(k-1)，

[0165] x3(k)=a31*x1(k-1)+a32*x2(k-1)+a33*x3(k-1)+$l31$ *w1(k-1)+$l32$ *w2(k-1)+$l33$ * w3(k-1)⇒

[0166] x1(k)＝x1(k-1)+1.0e-03*x2(k-1)+5.0e-07*x3(k-1)+w1(k-1)，

[0167] x2(k)＝+x2(k-1)+1.0e-03*x3(k-1)+w2(k-1)，

[0168] x3(k)=x3(k-1)+ w3(k-1)⇒

[0169] (A)如上所述在本示例中，OESMk完成Z(k)＝H*X(k)+V(k)，表示成矩阵 形式

[0170] $$\begin{bmatrix} z1 \\ \vdots \\ zn \end{bmatrix}_k = \begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix} * \begin{bmatrix} x1 \\ \vdots \\ xn \end{bmatrix}_k + \begin{bmatrix} v1 \\ \vdots \\ vn \end{bmatrix}_k \Rightarrow \begin{bmatrix} z1 \\ z2 \\ z3 \end{bmatrix}_k = [h1\ h2\ h3] * \begin{bmatrix} x1 \\ x2 \\ x3 \end{bmatrix}_k + \begin{bmatrix} v1 \\ v2 \\ v3 \end{bmatrix}_k \Rightarrow \begin{bmatrix} z1 \\ z2 \\ z3 \end{bmatrix}_k = [1 \quad Ts/2 \quad Ts\hat{}2/6] * \begin{bmatrix} x1 \\ x2 \\ x3 \end{bmatrix}_k + \begin{bmatrix} v1 \\ 0 \\ 0 \end{bmatrix}_k \Rightarrow \begin{bmatrix} z1 \\ z2 \\ z3 \end{bmatrix}_k = [1 \quad 5.0e-04 \quad 1.67e-07] * \begin{bmatrix} x1 \\ x2 \\ x3 \end{bmatrix}_k + \begin{bmatrix} v1 \\ v2 \\ v3 \end{bmatrix}_k,$$

[0171] 表示成方程形式为，

[0172] z(k)=h0*x1(k)+h1*x2(k)+h2*x3(k)+v(k) ⇒z(k)=x1(k)+$Ts/2$*x2(k)+$Ts\hat{}2/6$*x3(k)+v(k) ⇒z(k)=x1(k)+$5.0e-04$*x2(k)+$1.67e-07$*x3(k)+v(k)

[0173] (B)如上所述在本示例中，SPESMk完成X_pre(k)＝A*Xkf(k-1)，表示成 矩阵形式为，$$\begin{bmatrix} Xpre1 \\ \vdots \\ Xpren \end{bmatrix}_k = \begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix} * \begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_{k-1} \Rightarrow \begin{bmatrix} Xpre1 \\ Xpre2 \\ Xpre3 \end{bmatrix}_k = \begin{bmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{bmatrix} * \begin{bmatrix} Xkf1 \\ Xkf2 \\ Xkf3 \end{bmatrix}_{k-1} \Rightarrow \begin{bmatrix} Xpre1 \\ Xpre2 \\ Xpre3 \end{bmatrix}_k = \begin{bmatrix} 1 & Ts & Ts\hat{}2/2 \\ 0 & 1 & Ts \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} Xkf1 \\ Xkf2 \\ Xkf3 \end{bmatrix}_{k-1} \Rightarrow \begin{bmatrix} Xpre1 \\ Xpre2 \\ Xpre3 \end{bmatrix}_k = \begin{bmatrix} 1 & 1.0e-03 & 5.0e-07 \\ 0 & 1 & 1.0e-03 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} Xkf1 \\ Xkf2 \\ Xkf3 \end{bmatrix}_{k-1}$$ 表示成方程 形式为，

[0174] X_pre1(k)＝a11*Xkf1(k-1)+a12*Xkf2(k-1)+a13*Xkf3(k-1)，

[0175] X_pre2(k)＝a21*Xkf1(k-1)+a22*Xkf2(k-1)+a23*Xkf3(k-1)，

[0176] $Xpre3$(k)=a31* Xkf1(k-1)+a32* Xkf2(k-1)+a33* Xkf3(k-1) ⇒

[0177] X_pre1(k)＝Xkf1(k-1)+1.0e-03*Xkf2(k-1)+5.0e-07*Xkf3(k-1)，

[0178] X_pre2(k)＝Xkf2(k-1)+1.0e-03*Xkf3(k-1)，

[0179] $Xpre3$(k)=Xkf3(k-1) ⇒

[0180] (C)如上所述在本示例中，CPESMk完成计算方程为 P_pre(k)＝A*P(k-1)*A'+Q，表示成矩阵形式为，

$$\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k = \begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix} * \begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_{k-1} * \begin{bmatrix} a11 & \cdots & an1 \\ \vdots & \ddots & \vdots \\ a1n & \cdots & ann \end{bmatrix} + \begin{bmatrix} q11 & \cdots & q1n \\ \vdots & \ddots & \vdots \\ qn1 & \cdots & qnn \end{bmatrix} \Rightarrow$$

[0181] $$\begin{bmatrix} P_pre11 & P_pre12 & P_pre13 \\ P_pre21 & P_pre22 & P_pre23 \\ P_pre31 & P_pre32 & P_pre33 \end{bmatrix}_k = \begin{bmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{bmatrix} * \begin{bmatrix} P11 & P12 & P13 \\ P21 & P22 & P23 \\ P31 & P32 & P33 \end{bmatrix}_{k-1} * \begin{bmatrix} a11 & a21 & a31 \\ a12 & a22 & a32 \\ a13 & a23 & a33 \end{bmatrix} + \begin{bmatrix} q11 & q12 & q13 \\ q21 & q22 & q23 \\ q31 & q32 & q33 \end{bmatrix} \Rightarrow \begin{bmatrix} P_pre11 & P_pre12 & P_pre13 \\ P_pre21 & P_pre22 & P_pre23 \\ P_pre31 & P_pre32 & P_pre33 \end{bmatrix}_k = \begin{bmatrix} 1 & Ts & Ts\hat{}2/2 \\ 0 & 1 & Ts \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} P11 & P12 & P13 \\ P21 & P22 & P23 \\ P31 & P32 & P33 \end{bmatrix}_{k-1} * \begin{bmatrix} 1 & 0 & 0 \\ Ts & 1 & 0 \\ Ts\hat{}2/2 & Ts & 1 \end{bmatrix} + detav\hat{}2 * \begin{bmatrix} Ts\hat{}6/252 & Ts\hat{}5/72 & Ts\hat{}4/30 \\ Ts\hat{}5/72 & Ts\hat{}4/20 & Ts\hat{}3/8 \\ Ts\hat{}4/30 & Ts\hat{}3/8 & Ts\hat{}2/3 \end{bmatrix} \Rightarrow \begin{bmatrix} P_pre11 & P_pre12 & P_pre13 \\ P_pre21 & P_pre22 & P_pre23 \\ P_pre31 & P_pre32 & P_pre33 \end{bmatrix}_k = \begin{bmatrix} 1 & 1.0e-03 & 5.0e-07 \\ 0 & 1 & 1.0e-03 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} P11 & P12 & P13 \\ P21 & P22 & P23 \\ P31 & P32 & P33 \end{bmatrix}_{k-1} * \begin{bmatrix} 1 & 0 & 0 \\ 1.0e-03 & 1 & 0 \\ 5.0e-07 & 1.0e-03 & 1 \end{bmatrix} + \begin{bmatrix} 3.97e-27 & 1.39e-23 & 3.33e-20 \\ 1.39e-23 & 5.0e-20 & 1.25e-16 \\ 3.33e-20 & 1.25e-16 & 3.33e-13 \end{bmatrix}$$

[0182] 表示成方程形式为，

[0183] P_pre11(k)＝

[0184] (A11*P11(k-1)+A12*P21(k-1)+A13*P31(k-1))*A11+(A11*P12(k-1)+A12*P22(k-1)+A13*P32(k-1))*A12+(A11*P13(k-1)+A12*P23(k-1)+A13*P33(k-1))*A13+Q 11；

[0185] P_pre12(k)＝

[0186] (A11*P11(k-1)+A12*P21(k-1)+A13*P31(k-1))*A21+(A11*P12(k-1)+A12*P22(k-1)+A13*P32(k-1))*A22+(A11*P13(k-1)+A12*P23(k-1)+A13*P33(k-1))*A23+Q 12；

[0187] P_pre13(k)＝

[0188] (A11*P11(k-1)+A12*P21(k-1)+A13*P31(k-1))*A31+(A11*P12(k-1)+A12*P22(k-1)+A13*P32(k-1))*A32+(A11*P13(k-1)+A12*P23(k-1)+A13*P33(k-1))*A33+Q 13；

[0189] P_pre21(k)＝

[0190] (A21*P11(k-1)+A22*P21(k-1)+A23*P31(k-1))*A11+(A21*P12(k-1)+A22*P22(k-1)+A23*P32(k-1))*A12+(A21*P13(k-1)+A22*P23(k-1)+A23*P33(k-1))*A13+Q 21；

[0191] P_pre22(k)＝

[0192] (A21*P11(k-1)+A22*P21(k-1)+A23*P31(k-1))*A21+(A21*P12(k-1)+A22*P22(k-1)+A23*P32(k-1))*A22+(A21*P13(k-1)+A22*P23(k-1)+A23*P33(k-1))*A23+Q 22；

[0193] P_pre23(k)＝

[0194] (A21*P11(k-1)+A22*P21(k-1)+A23*P31(k-1))*A31+(A21*P12(k-1)+A22*P22(k-1)+A23*P32(k-1))*A32+(A21*P13(k-1)+A22*P23(k-1)+A23*P33(k-1))*A33+Q 23；

[0195] P_pre31(k)＝

[0196] (A31*P11(k-1)+A32*P21(k-1)+A33*P31(k-1))*A11+(A31*P12(k-1)+A32*P22(k-1)+A33*P32(k-1))*A12+(A31*P13(k-1)+A32*P23(k-1)+A33*P33(k-1))*A13+Q 31；

[0197] P_pre32(k)＝

[0198] (A31*P11(k-1)+A32*P21(k-1)+A33*P31(k-1))*A21+(A31*P12(k-1)+A32*P22(k-1)+A33*P32(k-1))*A22+(A31*P13(k-1)+A32*P23(k-1)+A33*P33(k-1))*A23+Q 32；

[0199] P_pre33(k)＝

[0200] (A31*P11(k-1)+A32*P21(k-1)+A33*P31(k-1))*A31+(A31*P12(k-1)+A32*P22(k-1)+A33*P32(k-1))*A32+(A31*P13(k-1)+A32*P23(k-1)+A33*P33(k-1))*A33+Q 33；

[0201] (D)如上所述在本示例中，KGESMk完成计算方程为 Kg(k)＝P_pre(k)*H′*inv(H*P_pre(k)*H′+R)。表示成矩阵形式为，

$$\begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_k = \begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k * \begin{bmatrix} h11 & \cdots & hm1 \\ \vdots & \ddots & \vdots \\ h1n & \cdots & hmn \end{bmatrix} *\mathrm{inv}\left(\begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hm1 & \cdots & hmn \end{bmatrix} * \begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k * \begin{bmatrix} h11 & \cdots & hm1 \\ \vdots & \ddots & \vdots \\ h1n & \cdots & hmn \end{bmatrix} + R \right) \Rightarrow \begin{bmatrix} Kg1 \\ Kg2 \\ Kg3 \end{bmatrix}_k = \begin{bmatrix} P_pre11 & P_pre12 & P_pre13 \\ P_pre21 & P_pre22 & P_pre23 \\ P_pre31 & P_pre32 & P_pre33 \end{bmatrix}_k * \begin{bmatrix} h1 \\ h2 \\ h3 \end{bmatrix}$$

$$*\mathrm{inv}\left([h1\ h2\ h3] * \begin{bmatrix} P_pre11 & P_pre12 & P_pre13 \\ P_pre21 & P_pre22 & P_pre23 \\ P_pre31 & P_pre32 & P_pre33 \end{bmatrix}_k * \begin{bmatrix} h1 \\ h2 \\ h3 \end{bmatrix} + R \right) \Rightarrow \begin{bmatrix} Kg1 \\ Kg2 \\ Kg3 \end{bmatrix}_k = \begin{bmatrix} P_pre11 & P_pre12 & P_pre13 \\ P_pre21 & P_pre22 & P_pre23 \\ P_pre31 & P_pre32 & P_pre33 \end{bmatrix}_k *$$

$$\begin{bmatrix} 1 \\ Ts/2 \\ Ts\hat{}2/6 \end{bmatrix} *\mathrm{inv}\left([1\ Ts/2\ Ts\hat{}2/6] * \begin{bmatrix} Ppre11 & Ppre12 & Ppre13 \\ Ppre21 & Ppre22 & Ppre23 \\ Ppre31 & Ppre32 & Ppre33 \end{bmatrix}_k * \begin{bmatrix} 1 \\ Ts/2 \\ Ts\hat{}2/6 \end{bmatrix} + detaw\hat{}2/Ts\hat{}4 \right) \Rightarrow \begin{bmatrix} Kg1 \\ Kg2 \\ Kg3 \end{bmatrix}_k =$$

$$\begin{bmatrix} P_pre11 & P_pre12 & P_pre13 \\ P_pre21 & P_pre22 & P_pre23 \\ P_pre31 & P_pre32 & P_pre33 \end{bmatrix}_k * \begin{bmatrix} 1 \\ 5.0e-04 \\ 1.67e-07 \end{bmatrix} *\mathrm{inv}\left([1\ 5.0e-04\ 1.67e-07] * \begin{bmatrix} Ppre11 & Ppre12 & Ppre13 \\ Ppre21 & Ppre22 & Ppre23 \\ Ppre31 & Ppre32 & Ppre33 \end{bmatrix}_k * \begin{bmatrix} 1 \\ 5.0e-04 \\ 1.67e-07 \end{bmatrix} *+(1.0e+06) \right)$$

[0202] 表示成方程形式为，令

[0203] kgd＝(P_pre11(k)*H1*H1+P_pre21(k)*H1*H2+P_pre31(k)*H1*H3+P_pre12(k)*H1*H2+P_pre22(k)*H2*H2+P_pre32(k)*H2*H3+P_pre13(k)*H1*H3+P_pre23(k)*H2*H3+P_pre33(k)*H3*H3)+R；

[0204] kgn1(k)＝P_pre11(k)*H1+P_pre12(k)*H2+P_pre13(k)*H3；

[0205] kgn2(k)＝P_pre21(k)*H1+P_pre22(k)*H2+P_pre23(k)*H3；

[0206] kgn3(k)＝P_pre31(k)*H1+P_pre32(k)*H2+P_pre33(k)*H3；

[0207] Kg1＝kgn1/kgd；

[0208] Kg2＝kgn2/kgd；

[0209] Kg3＝kgn3/kgd；⇒

[0210] kgd＝(P_pre11(k)+P_pre21(k)*(Ts/2)+P_pre31(k)*(Ts^2/6)+P_pre12(k)*(Ts/2)+P _pre22(k)*(Ts/2)*(Ts/2)+P_pre32(k)*(Ts/2)*(Ts^2/6)+P_pre13(k)*(Ts^2/6)+P_pre23(k) *(Ts/2)*(Ts^2/6)+P_pre33(k)*(Ts^2/6)*(Ts^2/6))+(detaw^2/Ts^4)；

[0211] kgn1(k)＝P_pre11(k)+P_pre12(k)*(Ts/2)+P_pre13(k)*(Ts^2/6)；

[0212] kgn2(k)＝P_pre21(k)+P_pre22(k)*(Ts/2)+P_pre23(k)*(Ts′2/6)；

[0213] kgn3(k)＝P_pre31(k)+P_pre32(k)*(Ts/2)+P_pre33(k)*(Ts^2/6)；

[0214] Kg1＝kgn1/kgd；

[0215] Kg2＝kgn2/kgd；

[0216] Kg3＝kgn3/kgd；⇒

[0217] kgd＝(P_pre11(k)+P_pre21(k)*(5.0e-04)+P_pre31(k)*(1.67e-07)+P_pre12(k)*(5.0e-04)+P_pre22(k)*(5.0e-04)*(5.0e-04)+P_pre32(k)*(5.0e-04)*(1.67e-07)+P_pre13 (k)*(1.67e-07)+P_pre23(k)*(5.0e-04)*(1.67e-07)+P_pre33(k)*(1.67e-07)*(1.67e-07))+(1.0e+06)；

[0218] kgn1(k)＝P_pre11(k)+P_pre12(k)*(5.0e-04)+P_pre13(k)*(1.67e-07)；

[0219] kgn2(k)＝P_pre21(k)+P_pre22(k)*(5.0e-04)+P_pre23(k)*(1.67e-07)；

[0220] kgn3(k)＝P_pre31(k)+P_pre32(k)*(5.0e-04)+P_pre33(k)*(1.67e-07)；

[0221] Kg1＝kgn1/kgd；

[0222] Kg2＝kgn2/kgd；

[0223] Kg3＝kgn3/kgd；

[0224] (E)如上所述在本示例中，SUESMk其计算方程为 Xkf(k)＝X_pre(k)+Kg(k)*(Z(k)-H*X_pre(k))。

[0225] 表示成矩阵形式为，

[0226]
$$\begin{bmatrix} Xkf1 \\ \vdots \\ Xkfn \end{bmatrix}_k = \begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_k + \begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_k * \left(\begin{bmatrix} z1 \\ \vdots \\ zn \end{bmatrix}_k - \begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix} * \begin{bmatrix} X_pre1 \\ \vdots \\ X_pren \end{bmatrix}_k \right)$$

$$\Rightarrow \begin{bmatrix} Xkf1 \\ Xkf2 \\ Xkf3 \end{bmatrix}_k = \begin{bmatrix} X_pre1 \\ X_pre2 \\ X_pre3 \end{bmatrix}_k + \begin{bmatrix} Kg1 \\ Kg2 \\ Kg3 \end{bmatrix}_k * \left(z(k) - [h1\ h2\ h3] * \begin{bmatrix} X_pre1 \\ X_pre2 \\ X_pre3 \end{bmatrix}_k \right) \Rightarrow \begin{bmatrix} Xkf1 \\ Xkf2 \\ Xkf3 \end{bmatrix}_k = \begin{bmatrix} X_pre1 \\ X_pre2 \\ X_pre3 \end{bmatrix}_k + \begin{bmatrix} Kg1 \\ Kg2 \\ Kg3 \end{bmatrix}_k * \left(z(k) - [1\ \ Ts/2\ \ Ts^2/6] * \begin{bmatrix} X_pre1 \\ X_pre2 \\ X_pre3 \end{bmatrix}_k \right) = \begin{bmatrix} X_pre1 \\ X_pre2 \\ X_pre3 \end{bmatrix}_k + \begin{bmatrix} Kg1 \\ Kg2 \\ Kg3 \end{bmatrix}_k * \left(z(k) - [1\ \ 5.0e-04\ \ 1.67e-07] * \begin{bmatrix} X_pre1 \\ X_pre2 \\ X_pre3 \end{bmatrix}_k \right)$$

[0227] 表示成方程形式为，

[0228] Xkf1(k)＝X_pre1(k)+Kg1*(Z(k)-H1*X_pre1(k)+H2*X_pre2(k)+H3*X_pre3(k))；

[0229] Xkf2(k)＝X_pre2(k)+Kg2*(Z(k)-H1*X_pre1(k)+H2*X_pre2(k)+H3*X_pre3(k))；

[0230] Xkf3(k)＝X_pre3(k)+Kg3*(Z(k)-H1*X_pre1(k)+H2*X_pre2(k)+H3*X_pre3(k))；⇒

[0231] Xkf1(k)＝X_pre1(k)+Kg1*(Z(k)-X_pre1(k)+(Ts/2)*X_pre2(k)+(Ts^2/6)*X_pre3(k))；

[0232] Xkf2(k)＝X_pre2(k)+Kg2*(Z(k)-X_pre1(k)+(Ts/2)*X_pre2(k)+(Ts^2/6)*X_pre3(k))；

[0233] Xkf3(k)＝X_pre3(k)+Kg3*(Z(k)-X_pre1(k)+(Ts/2)*X_pre2(k)+(Ts^2/6)*X_pre3(k))；⇒

[0234] Xkf1(k)＝X_pre1(k)+Kg1*(Z(k)-X_pre1(k)+(5.0e-04)*X_pre2(k)+(1.67e-07)*X_ pre3(k))；

[0235] Xkf2(k)＝X_pre2(k)+Kg2*(Z(k)-X_pre1(k)+(5.0e-04)*X_pre2(k)+(1.67e-07)*X_ pre3(k))；

[0236] Xkf3(k)＝X_pre3(k)+Kg3*(Z(k)-X_pre1(k)+(5.0e-04)*X_pre2(k)+(1.67e-07)*X_ pre3(k))；

[0237] (F)如上所述在本示例中，CUESMk其计算方程为 P(k)＝(I-Kg(k)*H)*P_pre(k)。表示成矩阵形式为，

[0238]

$$\begin{bmatrix} P11 & \cdots & P1n \\ \vdots & \ddots & \vdots \\ Pn1 & \cdots & Pnn \end{bmatrix}_k = \left(\begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix} - \begin{bmatrix} Kg11 & \cdots & Kg1n \\ \vdots & \ddots & \vdots \\ Kgn1 & \cdots & Kgnn \end{bmatrix}_k * \begin{bmatrix} h11 & \cdots & h1n \\ \vdots & \ddots & \vdots \\ hn1 & \cdots & hnn \end{bmatrix} \right) *$$

$$\begin{bmatrix} P_pre11 & \cdots & P_pre1n \\ \vdots & \ddots & \vdots \\ P_pren1 & \cdots & P_prenn \end{bmatrix}_k \Rightarrow \begin{bmatrix} p11 & p12 & p13 \\ p21 & p22 & p23 \\ p31 & p32 & p33 \end{bmatrix}_k = \left(\begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix} - \begin{bmatrix} Kg1 \\ Kg2 \\ Kg3 \end{bmatrix}_k * [h1\ h2\ h3] \right) * \begin{bmatrix} P_pre11 & P_pre12 & P_pre13 \\ P_pre21 & P_pre22 & P_pre23 \\ P_pre31 & P_pre32 & P_pre33 \end{bmatrix}_k$$

$$\Rightarrow \begin{bmatrix} p11 & p12 & p13 \\ p21 & p22 & p23 \\ p31 & p32 & p33 \end{bmatrix}_k = \left(\begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix} - \begin{bmatrix} Kg1 \\ Kg2 \\ Kg3 \end{bmatrix}_k * [1\ \ Ts/2\ \ Ts^2/6] \right) * \begin{bmatrix} P_pre11 & P_pre12 & P_pre13 \\ P_pre21 & P_pre22 & P_pre23 \\ P_pre31 & P_pre32 & P_pre33 \end{bmatrix}_k$$

$$\Rightarrow \begin{bmatrix} p11 & p12 & p13 \\ p21 & p22 & p23 \\ p31 & p32 & p33 \end{bmatrix}_k = \left(\begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix} - \begin{bmatrix} Kg1 \\ Kg2 \\ Kg3 \end{bmatrix}_k * [1\ \ 5.0e-04\ \ 1.67e-07] \right) * \begin{bmatrix} P_pre11 & P_pre12 & P_pre13 \\ P_pre21 & P_pre22 & P_pre23 \\ P_pre31 & P_pre32 & P_pre33 \end{bmatrix}_k$$

[0239] 表示成方程形式为，

[0240] P11(k)＝(-1)＊((Kg1＊H1-1)＊P_pre11(k)+Kg1＊H2＊P_pre21(k)+Kg1＊H3＊P_pre31(k))；

[0241] P12(k)＝(-1)＊((Kg1＊H1-1)＊P_pre12(k)+Kg1＊H2＊P_pre22(k)+Kg1＊H3＊P_pre3 2(k))；

[0242] P13(k)＝(-1)＊((Kg1＊H1-1)＊P_pre13(k)+Kg1＊H2＊P_pre23(k)+Kg1＊H3＊P_pre3 3(k))；

[0243] P21(k)＝(-1)＊(Kg2＊H1＊P_pre11(k)+(Kg2＊H2-1)＊P_pre21(k)+Kg2＊H3＊P_pre3 1(k))；

[0244] P22(k)＝(-1)＊(Kg2＊H1＊P_pre12(k)+(Kg2＊H2-1)＊P_pre22(k)+Kg2＊H3＊P_pre3 2(k))；

[0245] P23(k)＝(-1)＊(Kg2＊H1＊P_pre13(k)+(Kg2＊H2-1)＊P_pre23(k)+Kg2＊H3＊P_pre3 3(k))；

[0246] P31(k)＝(-1)＊(Kg3＊H1＊P_pre11(k)+Kg3＊H2＊P_pre21(k)+(Kg3＊H3-1)＊P_pre3 1(k))；

[0247] P32(k)＝(-1)＊(Kg3＊H1＊P_pre12(k)+Kg3＊H2＊P_pre22(k)+(Kg3＊H3-1)＊P_pre3 2(k))；

[0248] P33(k)＝(-1)＊(Kg3＊H1＊P_pre13(k)+Kg3＊H2＊P_pre23(k)+(Kg3＊H3-1)＊P_pre3 3(k))；⇒

[0249] P11(k)＝(-1)＊((Kg1-1)＊P_pre11(k)+Kg1＊(Ts/2)＊P_pre21(k)+Kg1＊(Ts^2/6)＊P_pre 31(k))；

[0250] P12(k)＝(-1)＊((Kg1-1)＊P_pre12(k)+Kg1＊(Ts/2)＊P_pre22(k)+Kg1＊(Ts^2/6)＊P_pre 32(k))；

[0251] P13(k)＝(-1)＊((Kg1-1)＊P_pre13(k)+Kg1＊(Ts/2)＊P_pre23(k)+Kg1＊(Ts^2/6)＊P_pre 33(k))；

[0252] P21(k)＝(-1)＊(Kg2＊P_pre11(k)+(Kg2＊(Ts/2)-1)＊P_pre21(k)+Kg2＊(Ts^2/6)＊P_pre 31(k))；

[0253] P22(k)＝(-1)＊(Kg2＊P_pre12(k)+(Kg2＊(Ts/2)-1)＊P_pre22(k)+Kg2＊(Ts^2/6)＊P_pre 32(k))；

[0254] P23(k)＝(-1)＊(Kg2＊P_pre13(k)+(Kg2＊(Ts/2)-1)＊P_pre23(k)+Kg2＊(Ts^2/6)＊

P_pre 33(k))；

[0255] P31(k)＝(-1)＊(Kg3*P_pre11(k)+Kg3*(Ts/2)*P_pre21(k)+(Kg3*(Ts^2/6)-1)＊P_pre 31(k))；

[0256] P32(k)＝(-1)＊(Kg3*P_pre12(k)+Kg3*(Ts/2)*P_pre22(k)+(Kg3*(Ts^2/6)-1)＊P_pre 32(k))；

[0257] P33(k)＝(-1)＊(Kg3*P_pre13(k)+Kg3*(Ts/2)*P_pre23(k)+(Kg3*(Ts^2/6)-1)＊P_pre 33(k))；⇒

[0258] P11(k)＝(-1)＊((Kg1-1)*P_pre11(k)+Kg1*(5.0e-04)*P_pre21(k)+Kg1*(1.67e-07)＊ P_pre31(k))；

[0259] P12(k)＝(-1)＊((Kg1-1)*P_pre12(k)+Kg1*(5.0e-04)*P_pre22(k)+Kg1*(1.67e-07)＊ P_pre32(k))；

[0260] P13(k)＝(-1)＊((Kg1-1)*P_pre13(k)+Kg1*(5.0e-04)*P_pre23(k)+Kg1*(1.67e-07)＊ P_pre33(k))；

[0261] P21(k)＝(-1)＊(Kg2*P_pre11(k)+(Kg2*(5.0e-04)-1)*P_pre21(k)+Kg2*(1.67e-07)＊ P_pre31(k))；

[0262] P22(k)＝(-1)＊(Kg2*P_pre12(k)+(Kg2*(5.0e-04)-1)*P_pre22(k)+Kg2*(1.67e-07)＊ P_pre32(k))；

[0263] P23(k)＝(-1)＊(Kg2*P_pre13(k)+(Kg2*(5.0e-04)-1)*P_pre23(k)+Kg2*(1.67e-07)＊ P_pre33(k))；

[0264] P31(k)＝(-1)＊(Kg3*P_pre11(k)+Kg3*(5.0e-04)*P_pre21(k)+(Kg3*(1.67e-07)-1)＊ P_pre31(k))；

[0265] P32(k)＝(-1)＊(Kg3*P_pre12(k)+Kg3*(5.0e-04)*P_pre22(k)+(Kg3*(1.67e-07)-1)＊ P_pre32(k))；

[0266] P33(k)＝(-1)＊(Kg3*P_pre13(k)+Kg3*(5.0e-04)*P_pre23(k)+(Kg3*(1.67e-07)-1)＊ P_pre33(k))；

[0267] 1)完成6轮运算，即用图6所示的并行卡尔曼滤波系统产生3000个系统状 态值。

[0268] 2)设定系统状态量的初始值Omega1即X(3,3000)＝80*10^(3)，状态最优估 算值的初始值Xkf(3,300)＝80*10^(3)，其他数值保持不变，完成1轮运算，即 用图6所示的并行卡尔曼滤波系统产生500个系统状态值。

[0269] 3)设定系统状态量的初始值Omega1即X(3,3500)＝0，状态最优估算值的 初始值Xkf(3,3500)＝0，其他数值保持不变，完成11轮运算，即用图6所示的 并行卡尔曼滤波系统产生5500个系统状态值。

[0270] 4)设定系统状态量的初始值Omega1即X(3,9000)＝-80*10^(3)，状态最优 估算值的初始值Xkf(3,3500)＝-80*10^(3)，其他数值保持不变，完成1轮运算， 即用图6所示的并行卡尔曼滤波系统产生500个系统状态值。

[0271] 5)设定系统状态量的初始值Omega1即X(3,9500)＝0，状态最优估算值的 初始值Xkf(3,3500)＝0，其他数值保持不变，完成12轮运算，即用图6所示的 并行卡尔曼滤波系统产生6000个系统状态值。

[0272] 6)设定系统状态量的初始值Omega1即X(3,15500)＝80*10^(3)，状态最优 估算值

的初始值Xkf(3,3500)＝80*10^(3)，其他数值保持不变，完成1轮运算，即用图6所示的并行卡尔曼滤波系统产生500个系统状态值。

[0273] 7)设定系统状态量的初始值Omega1即X(3,16000)＝0，状态最优估算值的 初始值Xkf(3,3500)＝0，其他数值保持不变，完成8轮运算，即用图6所示的 并行卡尔曼滤波系统产生4000个系统状态值。

[0274] 至此我们用图6所示的并行卡尔曼滤波系统实现了20000状态的载波跟踪。

[0275] 将图6所示的并行卡尔曼滤波系统计算生成的20000系统状态中的多普勒 频偏DetaTheta绘制成图，即得到图7。

[0276] 将图6所示的并行卡尔曼滤波系统计算生成的20000系统状态中的频偏的 变化率Omega0绘制成图，即得到图8。

[0277] 将图6所示的并行卡尔曼滤波系统计算生成的20000系统状态中的频偏变 化率的变化率Omega1绘制成图，即得到图9。

[0278] 将20000各系统状态中的测量错误messure_err_x(k)＝Z(k)-X(1,k)和卡尔曼 估计误差kalman_err_x(k)＝Xkf(1,k)-X(1,k)绘制成图，即得到图10。

[0279] 以上所述实施例仅是为充分说明本发明而所举的较佳的实施例，本发明的 保护范围不限于此。本技术领域的技术人员在本发明基础上所作的等同替代或 变换，均在本发明的保护范围之内。本发明的保护范围以权利要求书为准。

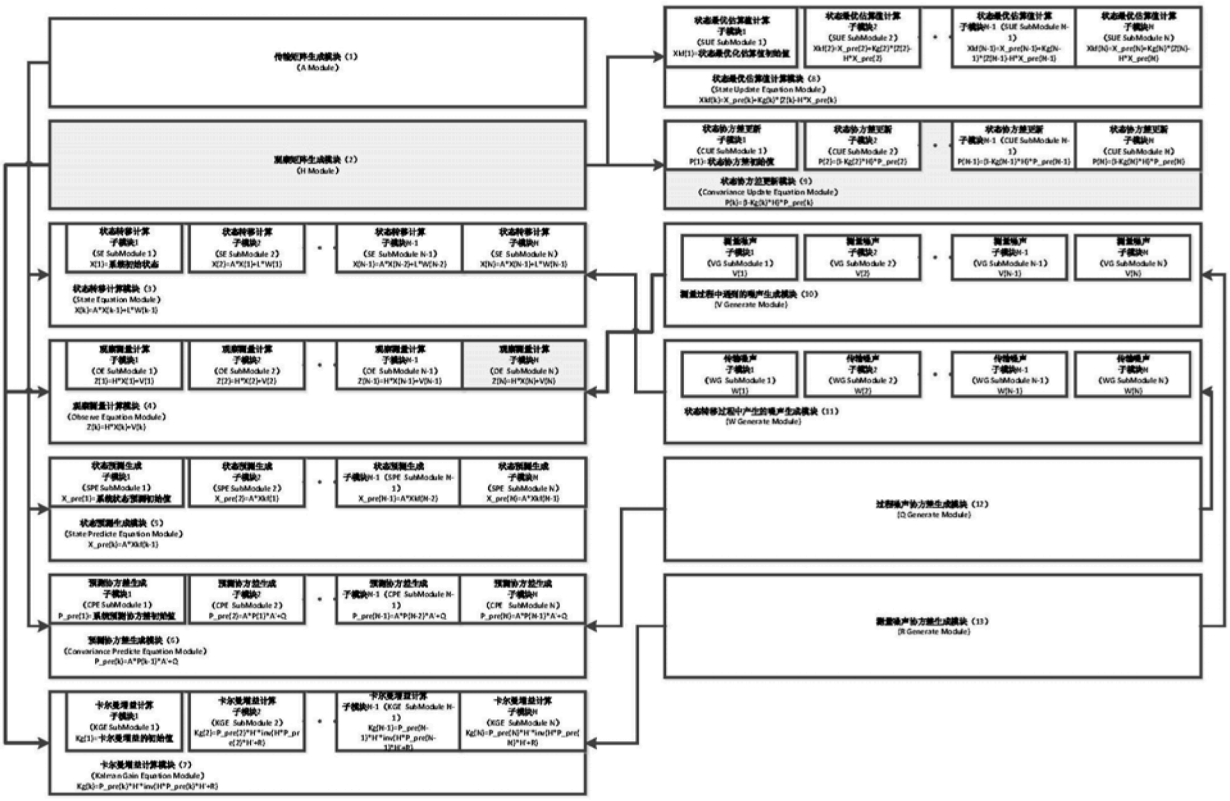

传感矩阵生成模块（1）
（A Module）
量测矩阵生成模块（2）
（H Module）
状态转移计算模块（3）
（State Equation Module）
X(k)=A*X(k-1)+L*W(k-1)
量测量计算模块（4）
（Observe Equation Module）
Z(k)=H*X(k)+V(k)
状态预测生成模块（5）
（State Predicte Equation Module）
X_pre(k)=A*Xkt(k-1)
预测协方差生成模块（6）
（Convergence Predicte Equation Module）
P_pre(k)=A*P(k-1)*A'+Q
卡尔曼增益计算模块（7）
（Kalman Gain Equation Module）
Kg(k)=P_pre(k)*H'*inv(H*P_pre(k)*H'+R)
状态最优估算值计算模块（8）
（State Update Equation Module）
Xkt(k)=X_pre(k)+Kg(k)*Z(k)-H*X_pre(k)
状态协方差更新模块（9）
（Convergence Update Equation Module）
P(k)=(I-Kg(k)*H)*P_pre(k)
量测过程中遇到的噪声生成模块（10）
（V Generate Module）
状态转移过程中产生的噪声生成模块（11）
（W Generate Module）
过程噪声协方差生成模块（12）
（Q Generate Module）
测量噪声协方差生成模块（13）
（R Generate Module）

图1

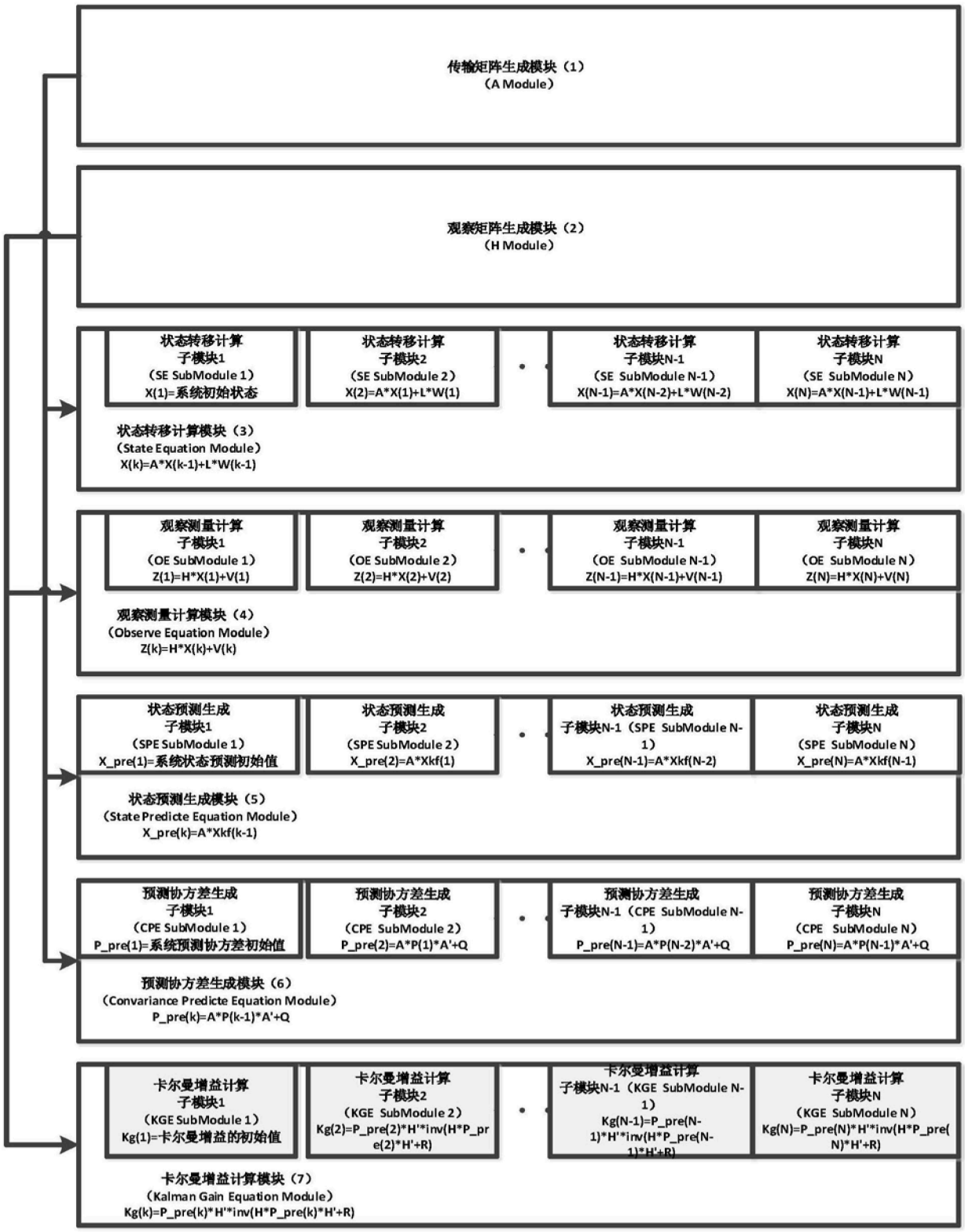

传输矩阵生成模块（1）
（A Module）
观察矩阵生成模块（2）
（H Module）
状态转移计算
子模块1
（SE SubModule 1）
X(1)=系统初始状态
状态转移计算
子模块2
（SE SubModule 2）
X(2)=A*X(1)+L*W(1)
状态转移计算
子模块N-1
（SE SubModule N-1）
X(N-1)=A*X(N-2)+L*W(N-2)
状态转移计算
子模块N
（SE SubModule N）
X(N)=A*X(N-1)+L*W(N-1)
状态转移计算模块（3）
（State Equation Module）
X(k)=A*X(k-1)+L*W(k-1)
观察测量计算
子模块1
（OE SubModule 1）
Z(1)=H*X(1)+V(1)
观察测量计算
子模块2
（OE SubModule 2）
Z(2)=H*X(2)+V(2)
观察测量计算
子模块N-1
（OE SubModule N-1）
Z(N-1)=H*X(N-1)+V(N-1)
观察测量计算
子模块N
（OE SubModule N）
Z(N)=H*X(N)+V(N)
观察测量计算模块（4）
（Observe Equation Module）
Z(k)=H*X(k)+V(k)
状态预测生成
子模块1
（SPE SubModule 1）
X_pre(1)=系统状态预测初始值
状态预测生成
子模块2
（SPE SubModule 2）
X_pre(2)=A*Xkf(1)
状态预测生成
子模块N-1（SPE SubModule N-1）
X_pre(N-1)=A*Xkf(N-2)
状态预测生成
子模块N
（SPE SubModule N）
X_pre(N)=A*Xkf(N-1)
状态预测生成模块（5）
（State Predicte Equation Module）
X_pre(k)=A*Xkf(k-1)
预测协方差生成
子模块1
（CPE SubModule 1）
P_pre(1)=系统预测协方差初始值
预测协方差生成
子模块2
（CPE SubModule 2）
P_pre(2)=A*P(1)*A'+Q
预测协方差生成
子模块N-1（CPE SubModule N-1）
P_pre(N-1)=A*P(N-2)*A'+Q
预测协方差生成
子模块N
（CPE SubModule N）
P_pre(N)=A*P(N-1)*A'+Q
预测协方差生成模块（6）
（Convariance Predicte Equation Module）
P_pre(k)=A*P(k-1)*A'+Q
卡尔曼增益计算
子模块1
（KGE SubModule 1）
Kg(1)=卡尔曼增益的初始值
卡尔曼增益计算
子模块2
（KGE SubModule 2）
Kg(2)=P_pre(2)*H'*inv(H*P_pre(2)*H'+R)
卡尔曼增益计算
子模块N-1（KGE SubModule N-1）
Kg(N-1)=P_pre(N-1)*H'*inv(H*P_pre(N-1)*H'+R)
卡尔曼增益计算
子模块N
（KGE SubModule N）
Kg(N)=P_pre(N)*H'*inv(H*P_pre(N)*H'+R)
卡尔曼增益计算模块（7）
（Kalman Gain Equation Module）
Kg(k)=P_pre(k)*H'*inv(H*P_pre(k)*H'+R)

图2

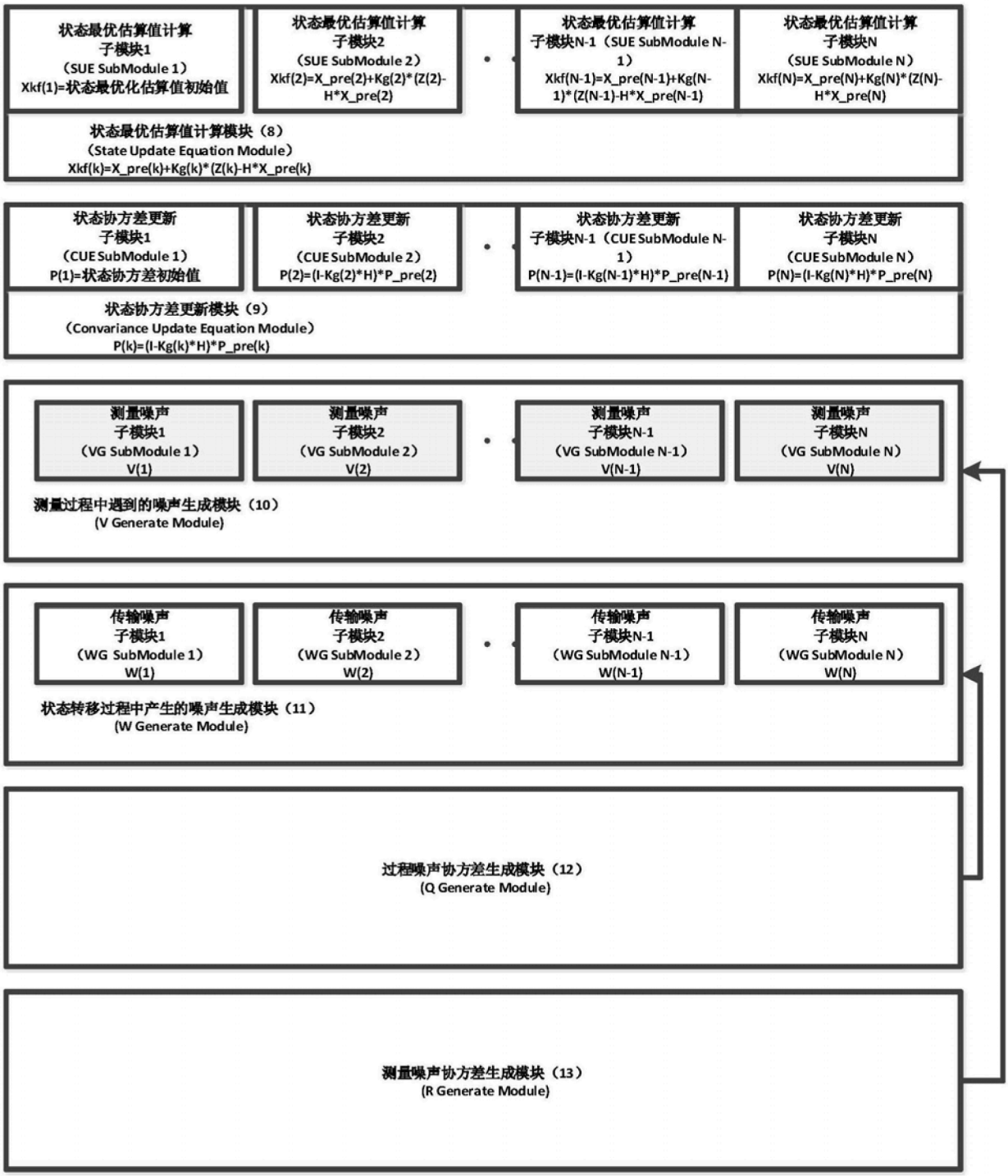

状态最优估算值计算
子模块1
（SUE SubModule 1）
Xkf(1)=状态最优化估算值初始值
状态最优估算值计算
子模块2
（SUE SubModule 2）
Xkf(2)=X_pre(2)+Kg(2)*(Z(2)-H*X_pre(2)
状态最优估算值计算
子模块N-1（SUE SubModule N-1）
Xkf(N-1)=X_pre(N-1)+Kg(N-1)*(Z(N-1)-H*X_pre(N-1)
状态最优估算值计算
子模块N
（SUE SubModule N）
Xkf(N)=X_pre(N)+Kg(N)*(Z(N)-H*X_pre(N)
状态最优估算值计算模块（8）
（State Update Equation Module）
Xkf(k)=X_pre(k)+Kg(k)*(Z(k)-H*X_pre(k)
状态协方差更新
子模块1
（CUE SubModule 1）
P(1)=状态协方差初始值
状态协方差更新
子模块2
（CUE SubModule 2）
P(2)=(I-Kg(2)*H)*P_pre(2)
状态协方差更新
子模块N-1（CUE SubModule N-1）
P(N-1)=(I-Kg(N-1)*H)*P_pre(N-1)
状态协方差更新
子模块N
（CUE SubModule N）
P(N)=(I-Kg(N)*H)*P_pre(N)
状态协方差更新模块（9）
（Convariance Update Equation Module）
P(k)=(I-Kg(k)*H)*P_pre(k)
测量噪声
子模块1
（VG SubModule 1）
V(1)
测量噪声
子模块2
（VG SubModule 2）
V(2)
测量噪声
子模块N-1
（VG SubModule N-1）
V(N-1)
测量噪声
子模块N
（VG SubModule N）
V(N)
测量过程中遇到的噪声生成模块（10）
(V Generate Module)
传输噪声
子模块1
（WG SubModule 1）
W(1)
传输噪声
子模块2
（WG SubModule 2）
W(2)
传输噪声
子模块N-1
（WG SubModule N-1）
W(N-1)
传输噪声
子模块N
（WG SubModule N）
W(N)
状态转移过程中产生的噪声生成模块（11）
(W Generate Module)
过程噪声协方差生成模块（12）
(Q Generate Module)
测量噪声协方差生成模块（13）
(R Generate Module)

图3

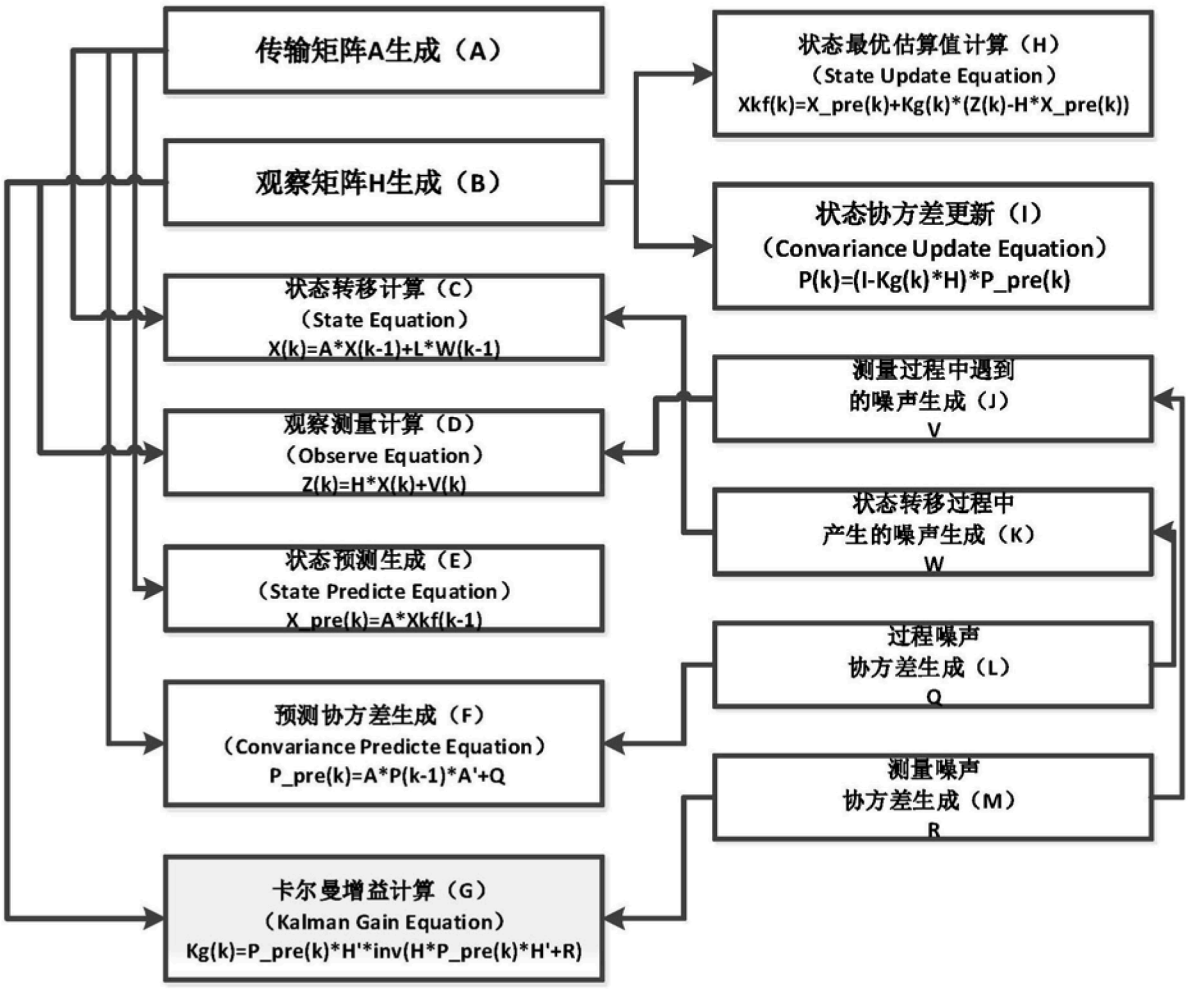

传输矩阵A生成（A）
观察矩阵H生成（B）
状态转移计算（C）
（State Equation）
X(k)=A*X(k-1)+L*W(k-1)
观察测量计算（D）
（Observe Equation）
Z(k)=H*X(k)+V(k)
状态预测生成（E）
（State Predicte Equation）
X_pre(k)=A*Xkf(k-1)
预测协方差生成（F）
（Convariance Predicte Equation）
P_pre(k)=A*P(k-1)*A'+Q
卡尔曼增益计算（G）
（Kalman Gain Equation）
Kg(k)=P_pre(k)*H'*inv(H*P_pre(k)*H'+R)
状态最优估算值计算（H）
（State Update Equation）
Xkf(k)=X_pre(k)+Kg(k)*(Z(k)-H*X_pre(k))
状态协方差更新（I）
（Convariance Update Equation）
P(k)=(I-Kg(k)*H)*P_pre(k)
测量过程中遇到
的噪声生成（J）
V
状态转移过程中
产生的噪声生成（K）
W
过程噪声
协方差生成（L）
Q
测量噪声
协方差生成（M）
R

图4

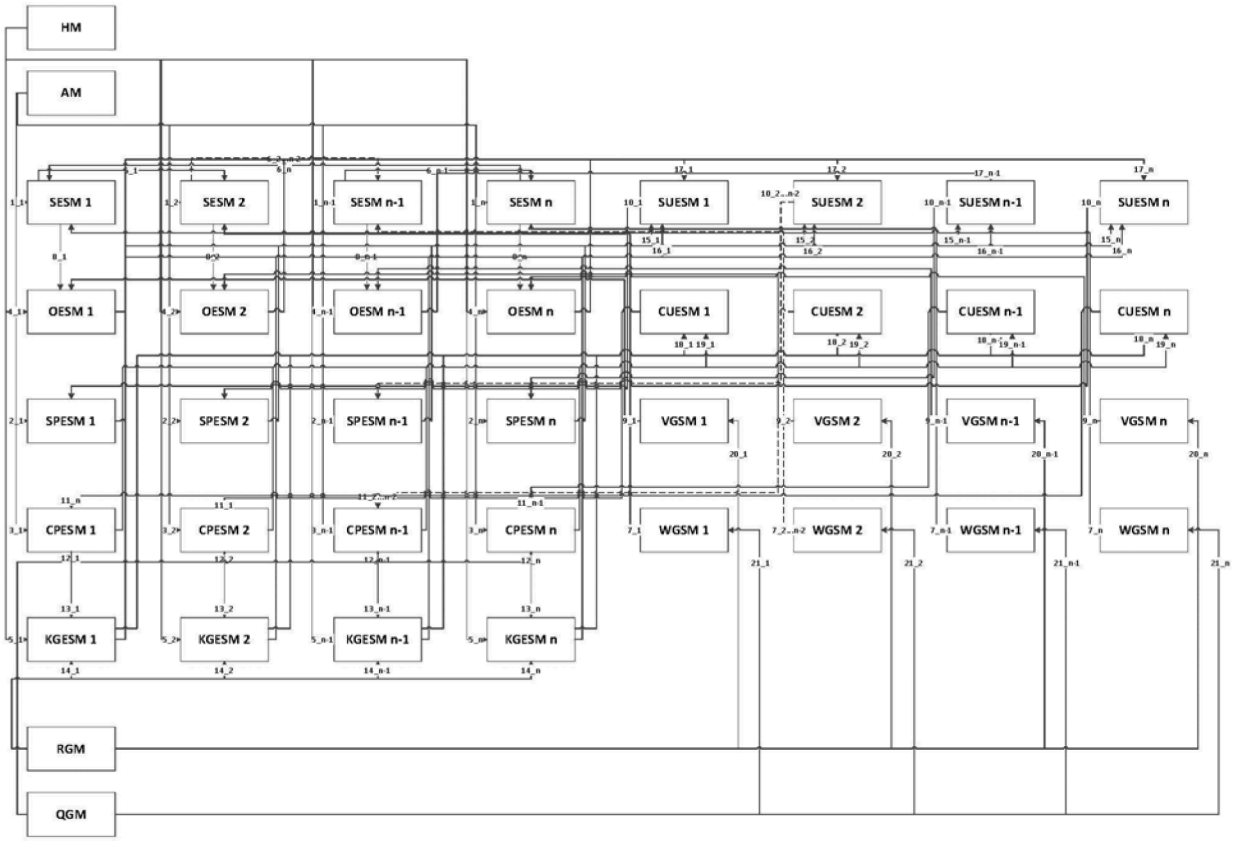

HM
AM
SESM 1
SESM 2
SESM n-1
SESM n
SUESM 1
SUESM 2
SUESM n-1
SUESM n
OESM 1
OESM 2
OESM n-1
OESM n
CUESM 1
CUESM 2
CUESM n-1
CUESM n
SPESM 1
SPESM 2
SPESM n-1
SPESM n
VGSM 1
VGSM 2
VGSM n-1
VGSM n
CPESM 1
CPESM 2
CPESM n-1
CPESM n
WGSM 1
WGSM 2
WGSM n-1
WGSM n
KGESM 1
KGESM 2
KGESM n-1
KGESM n
RGM
QGM

图5

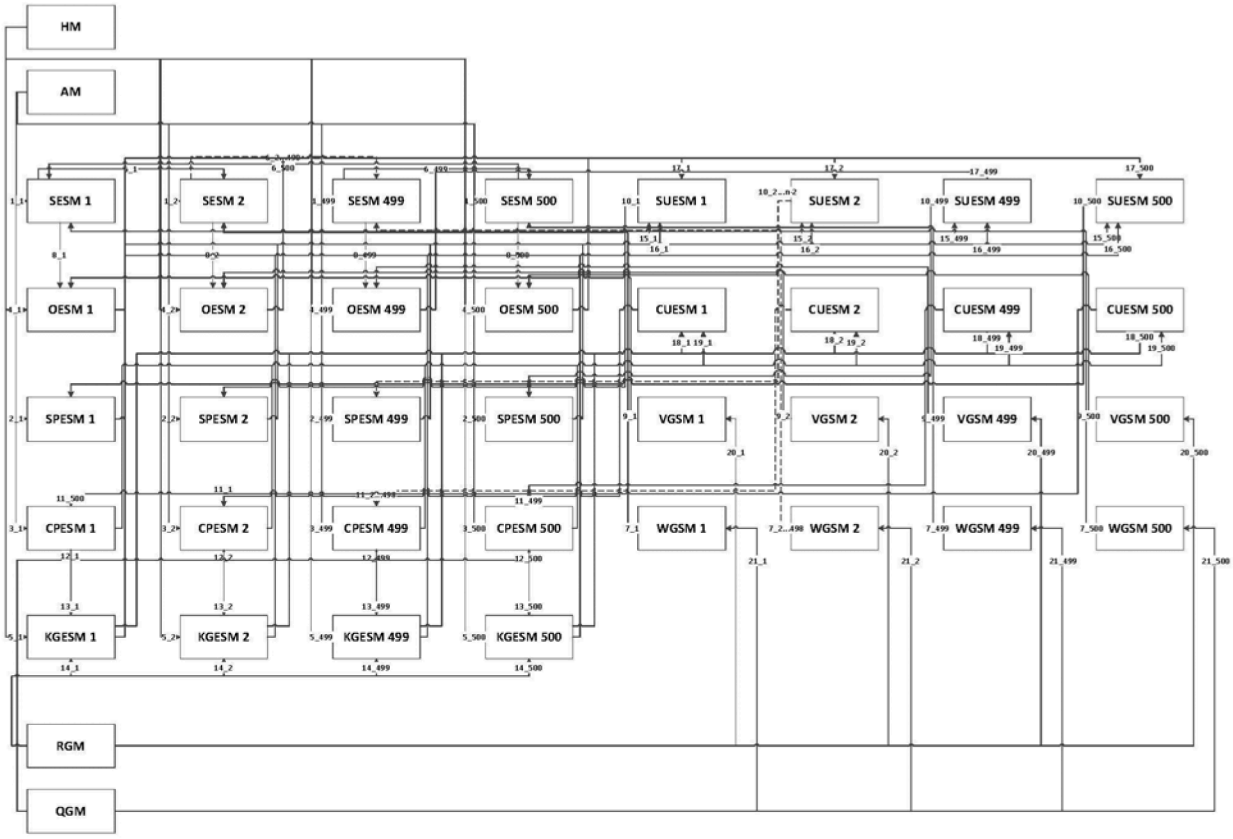

HM
AM
SESM 1
SESM 2
SESM 499
SESM 500
SUESM 1
SUESM 2
SUESM 499
SUESM 500
OESM 1
OESM 2
OESM 499
OESM 500
CUESM 1
CUESM 2
CUESM 499
CUESM 500
SPESM 1
SPESM 2
SPESM 499
SPESM 500
VGSM 1
VGSM 2
VGSM 499
VGSM 500
CPESM 1
CPESM 2
CPESM 499
CPESM 500
WGSM 1
WGSM 2
WGSM 499
WGSM 500
KGESM 1
KGESM 2
KGESM 499
KGESM 500
RGM
QGM

图6

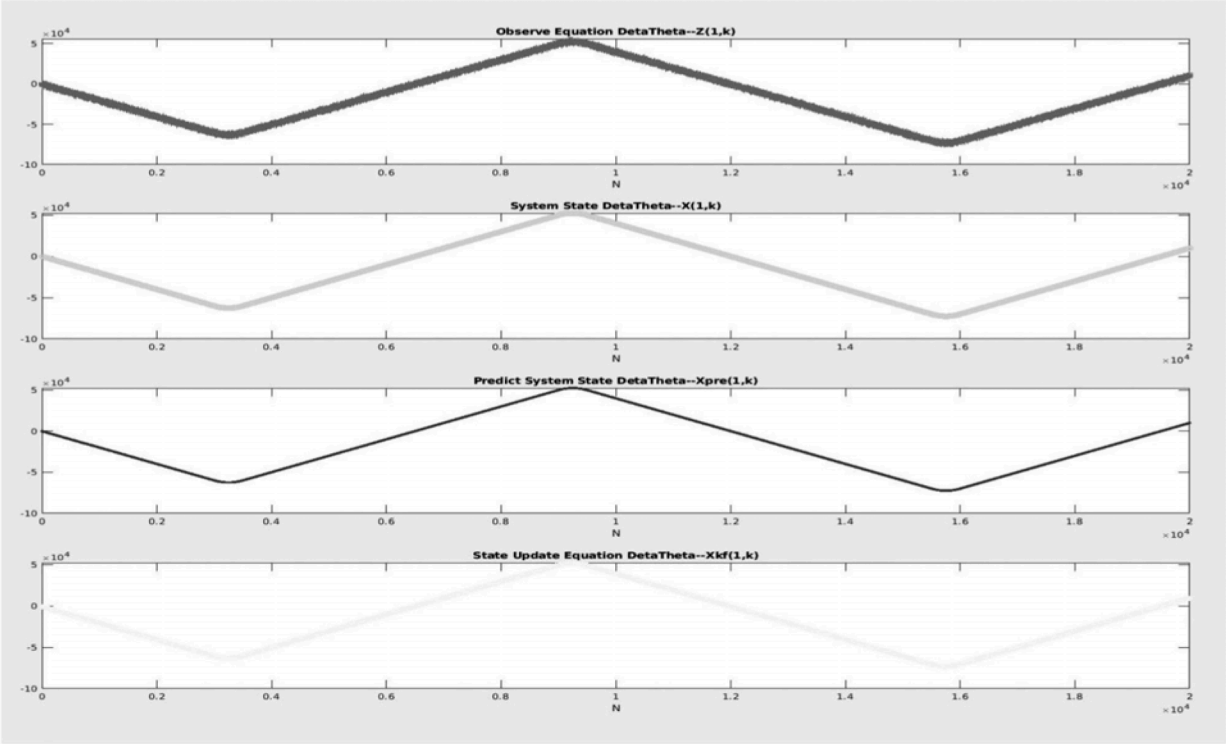

Observe Equation DetaTheta--Z(1,k)
System State DetaTheta--X(1,k)
Predict System State DetaTheta--Xpre(1,k)
State Update Equation DetaTheta--Xkf(1,k)
N

图7

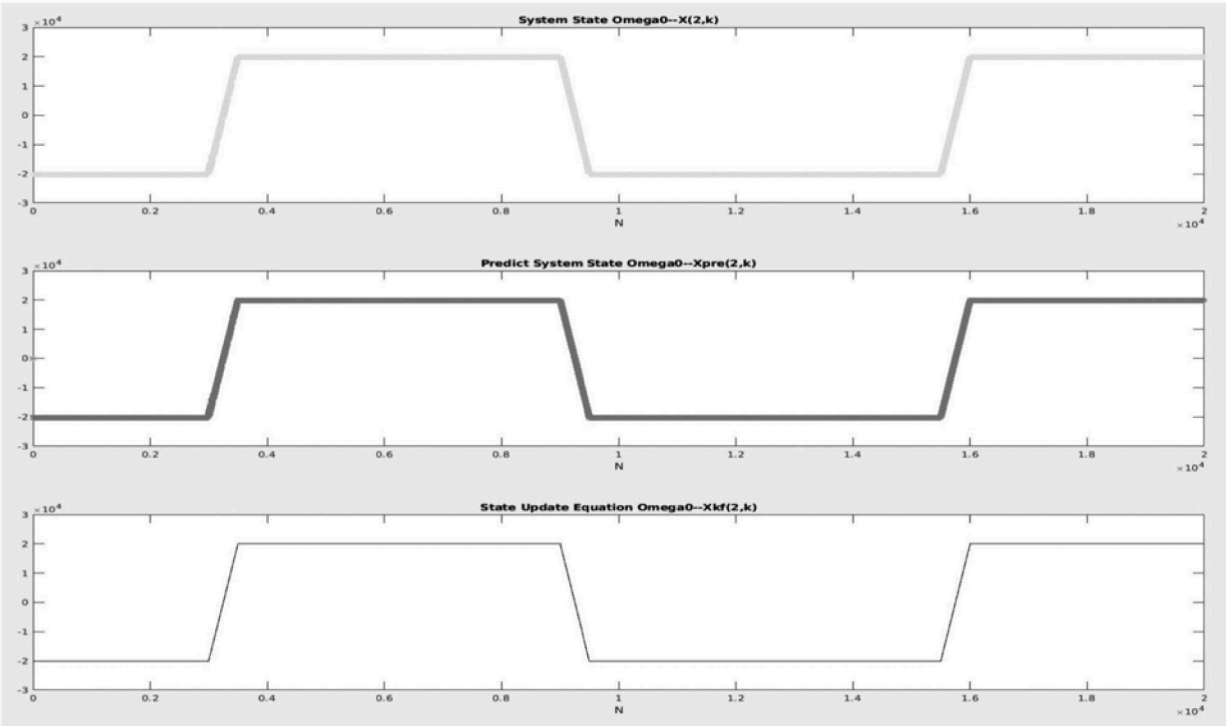

System State Omega0--X(2,k)
Predict System State Omega0--Xpre(2,k)
State Update Equation Omega0--Xkf(2,k)
N

图8

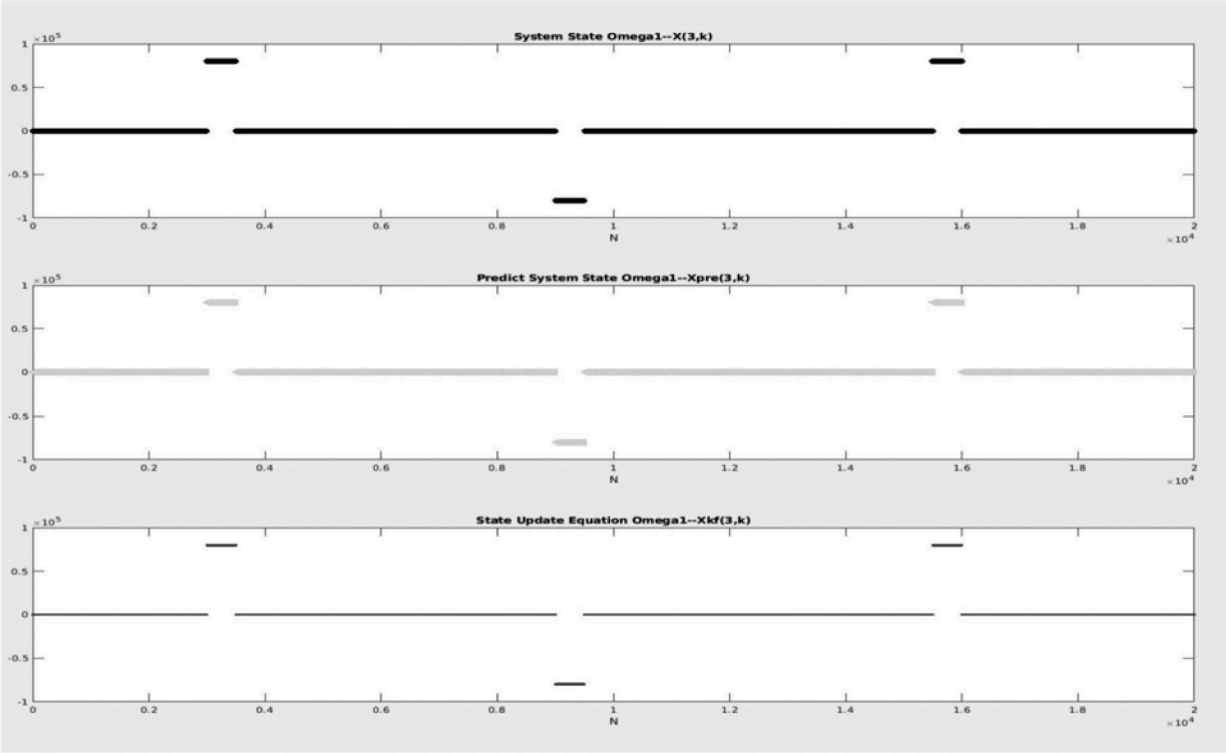

System State Omega1--X(3,k)
Predict System State Omega1--Xpre(3,k)
State Update Equation Omega1--Xkf(3,k)
N

图9

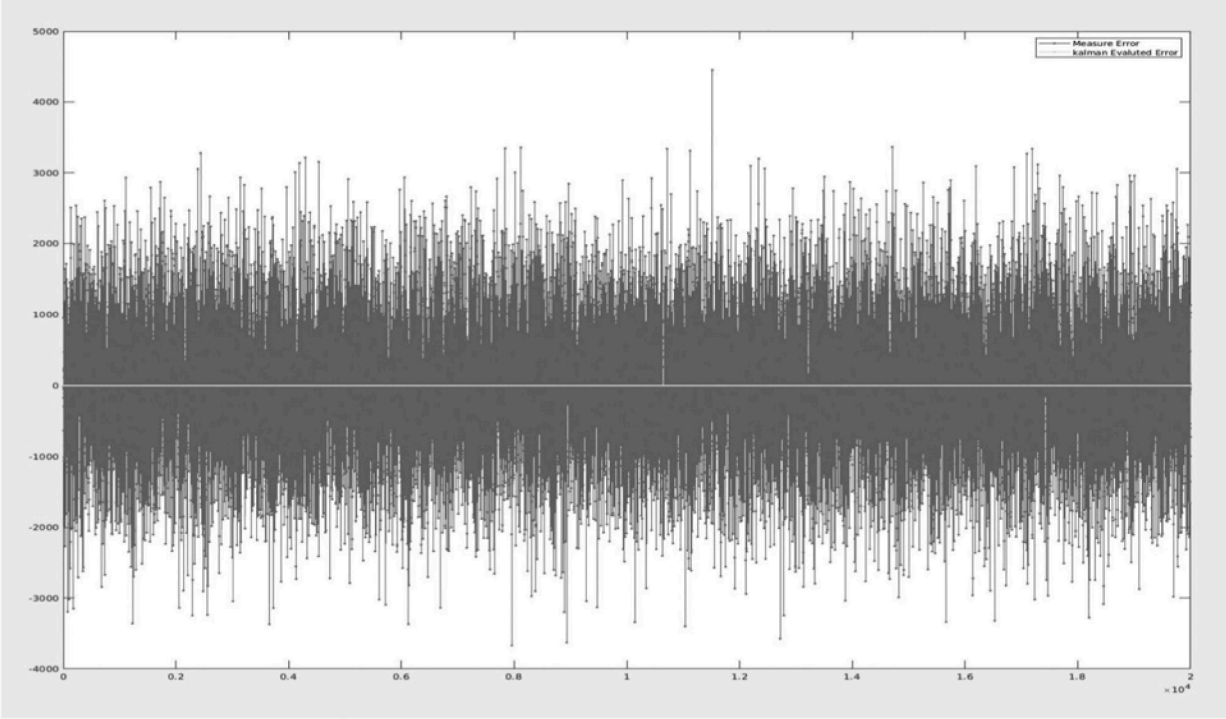

Measure Error
kalman Evaluted Error

图10