



Universidade Federal de Campina Grande  
Centro de Engenharia Elétrica e Informática  
Unidade Acadêmica de Engenharia Elétrica

Relatório do Trabalho de Conclusão de Curso:

# Sistema de Controle da Trajetória de um Robô Usando o Filtro de Kalman

**Maria Magdalena Ribeiro Cabral**

Campina Grande, Maio de 2007



Biblioteca Setorial do CDSA. Fevereiro de 2021.

Sumé - PB



Universidade Federal de Campina Grande  
Centro de Engenharia Elétrica e Informática  
Unidade Acadêmica de Engenharia Elétrica

## **Sistema de Controle da trajetória de um Robô usando Filtro de Kalman**

**Maria Magdalena Ribeiro Cabral**

Projeto de Conclusão de Curso  
submetida à Unidade Acadêmica de  
Engenharia Elétrica da Universidade  
Federal de Campina Grande para  
obtenção do grau de Bacharel em  
Engenharia Elétrica.

---

Prof. Dr. Hiran de Melo - Orientador

---

Prof. Dr. Raimundo Carlos Silvério Freire - Convidado

Campina Grande, maio 2007



***Dedicatória***

ao Meu Deus Eterno e Pai, como forma de louvor e agradecimento.

aos Meus Pais, *Apolônia e Saulo* por todo carinho, dedicação e apoio enquanto cursava esta graduação.

## AGRADECIMENTOS

À **DEUS**, por ser Fiel a cada momento, me dando forças para continuar mesmo em meio a tantas dificuldade.

Aos Professores **Hiran de Melo** e **Raimundo Carlos S. Freire**, por todo apoio que vêm me dando em todos esses anos de graduação, pela amizade e conselhos, além da confiança que sempre depositaram em mim, e pelo acompanhamento e orientação durante este projeto de conclusão de curso.

À minha Família em especial meus pais: **Saulo de Figueiredo Cabral** e **Apolônia Maria Ribeiro Cabral**, meus irmãos, por todo apoio, carinho e paciência que me deram durante todo o tempo em que eu estive nesta missão de trabalho.

Aos meus amigos, **Aretusa Kaline Alves Silva** e **Lincoln Reinaldo Silva**, pelo apoio nestes muitos anos de amizade, me estendendo à mão sempre que precisei.

Aos amigos: **Leonildo Pereira da Silva** e **José Ruben Sicchar**, que de maneira direta me auxiliaram para o desenvolvimento deste projeto.

À **José Aristides**, que me apoiou e incentivou durante todo o tempo em que esteve comigo. Aos amigos: **Jerônimo**, **Santhiago**, **Thierry**, **Thaís**, Prof. **Robert (Bob)** e **família**, **Eliete**, **Adail** e **Rosilda** que foram fundamentais me apoiando em muitos momentos, especialmente durante este curso.

Aos professores **Dr. Glauco Fontgalland** e **Dr. Marcos Antônio Barbosa de Melo**, pela confiança e oportunidades que me deram durante minha graduação.

### **Agradeço especialmente:**

À **UAEE – Unidade Acadêmica de Engenharia Elétrica da UFG** (Universidade Federal de Campina Grande), pela organização e base que me fizeram chegar aqui. Na pessoa do **Professor Mário Araújo Filho**.



## RESUMO

Baseado no avanço crescente da *robótica* na mecatrônica e da necessidade de técnicas de controle ótimo para robôs, onde uma localização exata é essencial para o desempenho correto das tarefas que estão à eles estão destinadas. E, sabendo que este problema não é de resolução elementar principalmente quando se pretender evitar custos elevados. Este trabalho apresenta o *Filtro de Kalman Discreto* (FKD) e o *Filtro de Kalman Estendido* (FKE) como técnicas de controle preditivo, implementado como um sistema estimador para correção da posição do Robô, para obtermos uma boa estimação da posição do robô, bem como, quando necessário, a correção de sua posição. Para exemplificar estas técnicas apresentamos uma trajetória linear para o FKD, e duas trajetórias - circular e elíptica - para o FKE, implementadas computacionalmente no Matlab<sup>®</sup>, onde conseguimos bons resultados. Além disso, apresentamos métodos de averiguação da validade do modelo e construção do filtro. Também demos enfoque ao fato de que para o FK, uma escolha não adequada dos parâmetros do sistema, bem como dos ruídos da observação e da estimação, pode gerar divergências no FK.

# Índice

Lista de Figuras e Quadros .....	2
1. Introdução .....	3
2. Fundamentação Teórica.....	5
2.1 Filtro de Kalman .....	5
2.2 Filtro de Kalman de tempo discreto .....	6
2.2.1. Pressupostos Básicos .....	6
2.2.2. Sistema e modelo de observação .....	8
2.2.3. Hipótese de Trabalho.....	9
2.2.4. Desenvolvimentos .....	9
2.2.5. Condição de não tendenciosidade .....	11
2.2.6. Determinando a covariância do erro.....	12
2.2.7 Escolhendo o Ganho de Kalman .....	12
2.2.8 Resumo das Equações Chaves.....	13
2.2.9 Interpretando o Filtro de Kalman .....	14
2.3 Desempenho do Filtro de Kalman .....	18
2.3.1 Desempenho sob condições de modelagem ideal .....	18
2.3.2 Validação do modelo.....	25
2.4 Filtro de Kalman Estendido.....	26
2.4.1 Predição .....	27
2.4.2 Predição e resíduo da observação.....	28
2.4.3 Atualização .....	29
2.4.4 Resumo das Equações .....	30
2.4.5 Divergência no Filtro.....	31
3 Implementação e Resultados .....	32
3.1 Modelo do Robô .....	32
3.2 Implementação do Filtro de Kalman no Matlab <sup>®</sup> para as Trajetórias Circular e Elíptica ...	35
4. Conclusão .....	41
5. Referências Bibliográficas.....	42

---



## Lista de Figuras e Quadros

Figura 01 – Modelo no Espaço de Estados .....	8
Figura 02 – Diagrama em Blocos do Filtro de Kalman Discreto no Tempo.....	14
Quadro 01 (a) – Código fonte do Matlab que implementa o Filtro de Kalman para representar a estimativa e observação de uma trajetória.....	16
Quadro 01(b) – Código fonte do Matlab que implementa o Filtro de Kalman para representar a estimativa e observação de uma trajetória.....	17
Figura 03 – Medidas (o) e estimativas (+) dos estados de uma trajetória.....	17
Figura 04: Trajetória da posição: a) Posições calculadas, previstas e observação; .....	19
Figura 04: b) trajetória inicial; c) trajetória em estado estacionário.....	19
Figura 05: trajetória da velocidade.....	19
Figura 06: (a) covariância do erro de posição; (b) covariância do erro de velocidade.....	20
Figure 07: (a) Efeito de usar valores de inicialização de posição diferentes;b) efeito de mudar R para inicializar $P_{00}$ .....	21
Figure 08: Resíduo e limites de desvio - padrão do resíduo.....	22
Figure 09: <i>Resíduo normalizado e média de movimento</i> .....	24
Figura 10: Auto-correlação para o resíduo.....	25
Figura 11: (a) Representação gráfica do estado associado ao Robô. (b) Geometria de um veículo-robô, explicitando o observador.....	32
Figura 12 – circunferência.....	35
Figura 13 – Elipse.....	35
Quadro 02 - Filtro de Kalman para controle de um robô em uma trajetória circular.....	36
Quadro 03 – Código que gera o estado do robô realizando uma trajetória circular.....	37
Figura 14 - Trajetória circular: Desejada (o), Observada (+) e Prevista pelo Filtro de Kalman (*), cortada antes de divergir.....	37
Quadro 04 – Filtro de Kalman para controle de um robô em uma trajetória elíptica.....	38
Quadro 05 – Código que gera o estado do robô realizando uma trajetória elíptica.....	39
Figura 15 – Trajetória Elíptica: Desejada (o), Observada(+) e Prevista pelo Filtro de Kalman (*), cortada antes de divergir.....	39
Figura 16 - Trajetória Elíptica: Desejada (azul), Observada (verde) e Prevista pelo Filtro de Kalman (vermelho), divergindo.....	40



## 1. Introdução

A mecatrônica inicialmente foi uma mistura das ciências mecânica e eletrônica, porém atualmente com o desenvolvimento científico e tecnológico atual, ela inclui outras ciências, entre elas, a eletrotécnica, a automação e a Informática.

O sistema automatizado de maior uso na mecatrônica é a manufatura integrada pelo computador que integra muitos equipamentos mecatrônicos na automação de todo o sistema de manufatura (robôs, CNCs, CLPs, etc). Dentre eles, a parte da mecatrônica mais desenvolvida nos últimos anos é a robótica.

A mecânica contribui com os componentes e a estrutura dos robôs, a eletricidade e eletrônica com seus sensores, atuadores e equipamento de controle, a automação com os algoritmos e estratégias de controle espacial e a informática com sua programação e supervisão. Assim, uma definição completa da Robótica poderia ser:

*“A robótica é uma ciência da engenharia aplicada que é tida como uma combinação da tecnologia de máquinas, de operadores e da ciência da computação. Inclui campos aparentemente tão diversos quanto projeto de máquinas, teoria de controle, microeletrônica, programação de computadores, inteligência artificial, fatores humanos e teoria da produção.”<sup>1</sup>*

Os robôs podem ser classificados em função do seu uso: Robôs Seqüenciais (2 a 4 graus de liberdade); Robôs Programáveis (4 a 6 grau de liberdade); Robôs Inteligentes (Robôs industriais especiais, Robôs móveis para aplicações espaciais, medicina, etc.), entre outros.

O uso dos robôs industriais tem diversas vantagens por seu amplo campo de aplicação, mas podem ser resumidas nos seguintes fatores:

- ✓ Fatores técnicos
  - Flexibilidade na gama de produtos fabricados.
  - Incremento da precisão, robustez, rapidez, uniformidade e suporte a ambientes hostis.
  - Incremento dos índices de qualidade e de peças rejeitadas.
- ✓ Fatores econômicos
  - Utilização eficiente de unidades de produção intensiva.
  - Aumento de produtividade (inexistência de interrupções, absentismos, etc.).
  - Redução do tempo do WIP (Work in Process) e do tempo de preparação da fábrica.

---

<sup>1</sup> Groover, Mikel P., Weiss, Mitchell. “Robótica: Tecnologia e Programação”. McGrawHill, São Paulo. 1989.



- ✓ • Fatores sociológicos
  - Redução do número de acidentes.
  - Afastamento do ser humano de locais perigosos para a saúde.
  - Redução de horários de trabalho.
  - Aumento do poder de compra.

Os avanços na automática que influem no desenvolvimento dos robôs e o incremento do uso dos mesmos nas indústrias, deve-se a alguns s fatores:

- ✓ . Robótica
  - – Redução no custo de 5-8%.
  - – Aumento no custo de mão de obra de 5%.
- ✓ . Tecnologia de Sensores
  - – Ganho em preço/desempenho de 10-15%.
- ✓ . Tecnologia de Controle
  - – Melhoria de 15-20%.

Na robótica móvel, e podemos incluir também os satélites, uma localização exata é essencial para o desempenho correto das tarefas que estão destinadas aos Robôs. Este problema não é de resolução elementar principalmente quando se pretender evitar custos elevados. Por esse motivo a localização de robôs móveis é motivo de uma intensa investigação científico-tecnológica.

Existe uma diversidade grande de técnicas baseadas em diferentes princípios físicos e diferentes algoritmos para localização. As técnicas baseadas em sistemas de visão apresentam, hoje em dia, custos muito interessantes e é possível a identificação de diversos elementos visuais da imagem.

Além destas, a implementação de algoritmos matemáticos estocásticos como meio de processamento de sinais discretos em sistemas de comunicação, tem permitido a melhoria da coordenação e o controle dos procedimentos de atividades de forma considerável. Assim, especificamente se observa que o uso de algoritmos inteligentes fundamentados no modelo matemático de filtros discretos de Kalman em sistemas de comunicação de robótica industrial, onde se tem como objetivo científico realizar a estimação, medição e localização da trajetória de movimentos realizados pelos mesmos, têm sido bastante satisfatórios.

Para esta problemática propomos implementar um sistema estimador para correção da posição do Robô baseado num Filtro de Kalman Estendido. Com esta proposta espera-se como benefício uma boa estimação da posição do robô, bem como, quando necessário, a correção de sua posição.



## 2. Fundamentação Teórica

### 2.1 Filtro de Kalman

O Filtro de Kalman foi criado por Rudolph E. Kalman em 1960, o Filtro de Kalman foi desenvolvido inicialmente como uma solução recursiva para filtragem linear de dados discretos. Para isto, utiliza equações matemáticas que implementam um estimador preditivo de estados, buscando corrigir iterativamente a resposta de um determinado sistema através de múltiplas variáveis relacionadas a ele.

Suas áreas de aplicação são muito diversificadas, tais como: processamento de imagem, supervisores de eventos discretos, processamento de sinais, sistemas de inferência, etc. A estrutura original do filtro de Kalman está baseada no **modelo de espaço de estados** (MEE) desenvolvido, inicialmente, para sistemas lineares. Este modelo busca definir a relação entrada-saída de um sistema linear indiretamente, por meio de um conjunto de variáveis internas  $k \times$  denominadas estados. Os estados são influenciados por seus próprios valores passados e pelas entradas do sistema, que por sua vez influenciam as saídas do sistema.

O problema da estimação do estado de sistemas estocásticos dinâmicos, a partir de observações da saída afetadas por ruído, é de grande importância na engenharia. O interesse neste problema data de aproximadamente dois séculos atrás.

Em relação à formulação do Filtro de Kalman para sistemas lineares faremos a seguinte consideração: A estimativa do estado do sistema é corrigida quando se processa uma observação com base nos resíduos. Estes resíduos podem ser utilizados para medir o desempenho do filtro, como mostramos mais adiante.

O Filtro de Kalman pode ser derivado da régua de Bayes. Considere o sistema geral a seguir. Onde cada passo do sistema de estados evolui de acordo com:

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{v}$$

A idéia é fazer uma previsão do estado de acordo com a equação de diferença estocástica acima, podemos determinar a (função densidade de probabilidade) p.d.f. da previsão como:

$$\begin{aligned} p(x(k+1)|x(k)) &= N(\mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k), \mathbf{A}\mathbf{P}(k|k)) \\ p(x(k+1)|x(k)) &= N(\hat{\mathbf{x}}(k+1|k), \mathbf{P}(k+1|k)) \end{aligned}$$

Isto se torna a nossa previsão para o próximo passo. Agora fazemos uma medida e dela determinamos função da probabilidade:

$$p(z(k+1)|x(k+1)) = N(\mathbf{H}\hat{\mathbf{x}}(k+1|k), \mathbf{R})$$

Podemos agora computar a densidade posterior combinando com o valor previsto e a probabilidade. Nossos resultados precedentes nos mostram que este será normal, e terão média e covariância:

$$\mathbf{P}(k+1|k+1) = \left( \mathbf{P}(k+1|k)^{-1} + \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \right)^{-1}$$

$$\hat{\mathbf{x}}(k+1|k+1) = \mathbf{P}(k+1|k+1) \left[ \mathbf{P}(k+1|k)^{-1} \hat{\mathbf{x}}(k+1|k) + \mathbf{H}^T \mathbf{R}^{-1} z(k+1) \right]$$

que são as equações de atualização do Filtro de Kalman. A última é vista mais frequentemente na forma:

$$\hat{\mathbf{x}}(k+1|k+1) = \hat{\mathbf{x}}(k+1|k) + \mathbf{P}(k+1|k+1) \mathbf{H} \mathbf{R}^{-1} \left[ z(k+1) - \mathbf{H} \hat{\mathbf{x}}(k+1|k) \right]$$

## 2.2 Filtro de Kalman de tempo discreto

### 2.2.1. Pressupostos Básicos

O problema que nos estamos dispendo a resolver é a estimação continua de um conjunto de parâmetros cujos valores mudam com o tempo. A atualização é encontrada por um conjunto de observações ou medidas,  $z(t)$ , as quais contem informação do sinal de interesse  $x(t)$ . A tarefa do estimador é providenciar uma estimativa  $\hat{x}(t+\tau)$  em algum instante do tempo  $(t+\tau)$ . Se  $\tau > 0$ , nos temos um **filtro de previsão**, se o contrário, se  $\tau < 0$  teremos um **filtro de alisamento** (ou, um Filtro de suavização - *smoothing filter*) e se  $\tau = 0$ , a operação é simplesmente chamada de **filtragem**.

Lembramos que um **estimador não polarizado de variância mínima (MVUE)** é um estimador que não é polarizado e que minimiza o erro médio quadrático:

$$\hat{\mathbf{x}} = \arg \min_{\hat{\mathbf{x}}} E \left[ \|\hat{\mathbf{x}} - \mathbf{x}\|^2 | z \right] = E \left[ \mathbf{x} | z \right]$$

O Termo  $E \left[ \|\mathbf{x} - \hat{\mathbf{x}}\|^2 \right]$ , chamado de variância do erro, demonstrado pela matriz de covariância de erro  $E \left[ (\mathbf{x} - \hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}})^T \right]$ . Especificamente, o erro de covariância de erro da matriz  $E \left[ \|\mathbf{x} - \hat{\mathbf{x}}\|^2 \right] = \text{trace} E \left[ (\mathbf{x} - \hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}})^T \right]$ .



## Notações

As notações seguintes serão utilizadas:

Símbolo	Significado	Definição matemática	Avaliação
$\mathbf{x}_k$	Vetor estado do sistema no instante $t_k$ .	$\mathbf{x}_k = \Phi_k \mathbf{x}_{k-1} + \mathbf{w}_k$	Vetor de dimensão n.
$t_k$	Instante da k-ésima medida.	-----	Escalar.
$k$	Índice das medidas no tempo discreto	-----	Inteiro.
$\mathbf{z}_k$	Vetor de observações no instante $k$ .	$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k$	Vetor de dimensão m.
$\mathbf{H}_k$	Matriz transição de saída.	-----	Matriz mxn.
$\mathbf{R}_k$	Matriz Covariância de ruído da medição.	$\mathbf{R}_k = \mathbf{E}(\mathbf{v}_k \mathbf{v}_k^T)$	Matriz mxm.
$\mathbf{v}_k$	Vetor ruído da medição.	Variável aleatória com média gaussiana zero e com a Covariância da matriz $\mathbf{R}_k$ .	Vetor de dimensão n.
$\Phi_k$	Matriz de transição do sistema linear dinâmico do instante $t_{k-1}$ para $t_k$	-----	Matriz nxn.
$\mathbf{w}_k$	Vetor ruído do processo (ou sistema, ou planta).	Variável aleatória com média gaussiana zero e com a Covariância da matriz $\mathbf{Q}_k$ .	Matriz nxn.
$\mathbf{Q}_k$	Matriz Covariância do ruído do processo (ou sistema, ou planta).	$\mathbf{Q}_k = \mathbf{E}(\mathbf{w}_k \mathbf{w}_k^T)$	Matriz nxn.
$\mathbf{P}_k$	Matriz Covariância do erro de $\mathbf{x}_k$ condicionado à $Z_k$ .	$\mathbf{P}_k = \mathbf{E}[(\mathbf{x}_k - \hat{\mathbf{x}}_k)(\mathbf{x}_k - \hat{\mathbf{x}}_k)^T   Z_k]$	Vetor de dimensão n.
$\mathbf{M}_k$	Matriz Covariância do erro de $\mathbf{x}_k$ condicionado à $Z_{k-1}$ .	$\mathbf{M}_k = \mathbf{E}[(\mathbf{x}_k - \bar{\mathbf{x}}_k)(\mathbf{x}_k - \bar{\mathbf{x}}_k)^T   Z_{k-1}]$	Vetor de dimensão n.
$\hat{\mathbf{x}}_k$	Estimativa de $\mathbf{x}_k$ no instante $t_k$ condicionado à $Z_k$ .	$\hat{\mathbf{x}}_k = \mathbf{E}(\mathbf{x}_k   Z_k)$	Vetor de dimensão n.
$\hat{\mathbf{x}}_{k i}$	Estimativa de $\mathbf{x}$ no instante $k$ baseada sobre o instante $i$ , $k > i$ .	-----	Vetor de dimensão n.
$\tilde{\mathbf{x}}_{k k}$	Erro de estimativa: $\hat{\mathbf{x}}_{k k} - \mathbf{x}_k$ (notação til).	-----	Vetor de dimensão n.
$\mathbf{F}_k$	Matriz transição de estado.	-----	Matriz mxn.
$\mathbf{G}_k$	Matriz transição de entrada (controle).	-----	Matriz mxn.
$\mathbf{K}_k$	Matriz Ganho de Kalman.	-----	Matriz mxm.
$\nu_k$	Inovação (Resíduo) no instante $k$ .	-----	-----
$\mathbf{S}_k$	Matriz Covariância de inovação (Resíduo) no instante $k$ .	-----	-----
$\mathbf{F}_k$	Matriz transição de estado.	-----	-----
$\mathbf{G}_k$	Matriz transição de entrada (controle).	-----	-----
$Z_k$	Conjunto de observações até (e incluindo) o instante $k$ .	$Z_k = \{k_1, k_2, k_3, \dots, k_k\}$	Conjunto de Vetores de dimensão m.
$\mathbf{P}_o$	Matriz de Covariância do Erro de Covariância da medida inicial.	$\mathbf{P}_o = \mathbf{E}[(\mathbf{x}_o - \hat{\mathbf{x}}_o)(\mathbf{x}_o - \hat{\mathbf{x}}_o)^T]$	Matriz nxn.

### 2.2.2. Sistema e modelo de observação

Agora começamos a análise do Filtro de Kalman. Observe a Figura 01. Supomos que o sistema pode ser modelado por uma equação de transição de estado,

$$\mathbf{x}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{G}_k \mathbf{u}_k + \mathbf{w}_k \quad 2.1$$

Em que  $\mathbf{x}_k$  é estado do sistema no instante  $k$ ,  $\mathbf{u}_k$  é um vetor entrada de controle,  $\mathbf{w}_k$  é um ruído aditivo do sistema,  $\mathbf{G}_k$  é a matriz de transição de entrada e  $\mathbf{F}_k$  é a matriz de transição de estado.

Supomos que as observações do estado são feitas através de medidas do sistema, as quais são representadas por uma equação linear na forma,

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \quad 2.2$$

Em que  $\mathbf{z}_k$  é a observação ou medida feita no tempo  $k$ ,  $\mathbf{x}_k$  é estado no instante  $k$ ,  $\mathbf{H}_k$  é a matriz de observação e  $\mathbf{v}_k$  é o ruído aditivo da medição.

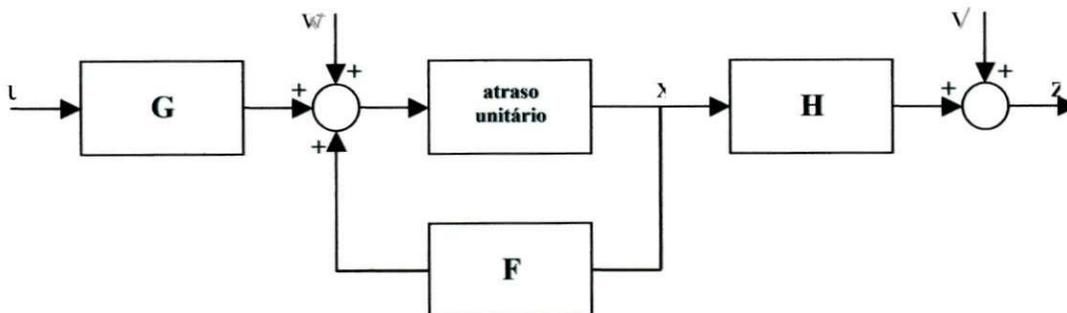


Figura 01 – Modelo no Espaço de Estados

### 2.2.3. Hipótese de Trabalho

Faremos as hipóteses seguintes:

- O ruído no sistema,  $\mathbf{w}_k$ , e o ruído na observação,  $\mathbf{v}_k$ , são processos aleatórios não correlacionados, brancos, com média zero e com covariâncias conhecidas. Então:

$$E[\mathbf{w}_k \mathbf{w}_l^T] = \begin{cases} \mathbf{Q}_k, & k = l \\ 0, & k \neq l \end{cases}; \quad 2.3$$

$$E[\mathbf{v}_k \mathbf{v}_l^T] = \begin{cases} \mathbf{R}_k, & k = l \\ 0, & k \neq l \end{cases}; \quad 2.4$$

$$E[\mathbf{w}_k \mathbf{v}_l^T] = 0, \text{ para todo } k, l. \quad 2.5$$

- Em que  $\mathbf{Q}_k$  e  $\mathbf{R}_k$  são matrizes simétricas positivas e semi-definidas.
- O estado inicial do sistema,  $x_0$ , é um vetor que é não correlacionado com ambos os ruídos.
- O estado inicial do sistema tem um valor médio conhecido e uma matriz de covariância.

$$\hat{x}_{0/0} = E[x_0] \text{ e } \mathbf{P}_{0/0} = E[(\hat{x}_{0/0} - x_0)(\hat{x}_{0/0} - x_0)^T] \quad 2.6$$

Dadas as hipóteses acima a tarefa é determinar, dado um conjunto de observações  $z_1, \dots, z_{k+1}$ , um filtro de estimação que no um dado instante de tempo seguinte,  $k+1$ , gere uma estimativa ótima do estado  $x_{k+1}$ , o qual será denotado por  $\hat{x}_{k+1}$ , que minimiza a expectância<sup>2</sup> do erro médio (função de custo/perdas),

$$E[\|x_{k+1} - \hat{x}_{k+1}\|^2] = E[(x_{k+1} - \hat{x}_{k+1})^T (x_{k+1} - \hat{x}_{k+1})] \quad 2.7$$

### 2.2.4. Desenvolvimentos

Considere a estimativa de estado  $\hat{x}_{k+1}$  baseadas sob as observações até o instante de tempo  $k$ ,  $z_1, \dots, z_{k+1}$ , isto é  $\hat{x}_{k+1/Z^k}$ . Esta estimativa é chamada de predição um passo à frente, ou simplesmente de uma **predição** (*Previsão*). Agora, a solução que minimiza a eq. (1.7) é a

---

<sup>2</sup> Esperança matemática.

expectância do estado no instante de tempo  $k+1$  condicionada as observações feitas até o instante  $k$ . Assim,

$$\hat{x}_{k+1|k} = E[x_{k+1} | z_1, \dots, z_k] = E[x_{k+1} | Z^k] \quad 2.8$$

Então a predição de estado é dada por

$$\begin{aligned} \hat{x}_{k+1|k} &= E[x_{k+1} | Z^k] = E[\mathbf{F}_k x_k + \mathbf{G}_k u_k + w_k | Z^k] \\ \hat{x}_{k+1|k} &= E[\mathbf{F}_k x_k | Z^k] + E[\mathbf{G}_k u_k | Z^k] + E[w_k | Z^k] \\ \hat{x}_{k+1|k} &= \mathbf{F}_k E[x_k | Z^k] + \mathbf{G}_k E[u_k | Z^k] + E[w_k | Z^k] \\ \hat{x}_{k+1|k} &= \mathbf{F}_k \hat{x}_{k|k} + \mathbf{G}_k u_k \end{aligned} \quad 2.9$$

Onde usamos o fato que o ruído do processo tem o valor médio zero e  $u_k$  é conhecido precisamente.

Observe que  $E[w_k | Z^k] = 0$  e  $E[u_k | Z^k] = u_k$ .

A variância da estimativa,  $\mathbf{P}_{k+1|k}$ , é o erro médio quadrático da estimativa  $\hat{x}_{k+1|k}$ :

Então, usando o fato de que  $w_k$  e  $\hat{x}_{k|k}$  não são correlacionados:

$$\begin{aligned} \mathbf{P}_{k+1|k} &= E[(x_{k+1} - \hat{x}_{k+1|k})(x_{k+1} - \hat{x}_{k+1|k})^T | Z^k] \\ \mathbf{P}_{k+1|k} &= E[(\mathbf{F}_k x_k + \mathbf{G}_k u_k + w_k - \mathbf{F}_k \hat{x}_{k|k} - \mathbf{G}_k u_k)(\mathbf{F}_k x_k + \mathbf{G}_k u_k + w_k - \mathbf{F}_k \hat{x}_{k|k} - \mathbf{G}_k u_k)^T | Z^k] \\ \mathbf{P}_{k+1|k} &= E[(\mathbf{F}_k x_k + w_k - \mathbf{F}_k \hat{x}_{k|k})(\mathbf{F}_k x_k + w_k - \mathbf{F}_k \hat{x}_{k|k})^T | Z^k] \\ \mathbf{P}_{k+1|k} &= E[(\mathbf{F}_k (x_k - \hat{x}_{k|k}) + w_k)(\mathbf{F}_k (x_k - \hat{x}_{k|k}) + w_k)^T | Z^k] \\ \mathbf{P}_{k+1|k} &= E[(\mathbf{F}_k (x_k - \hat{x}_{k|k}) + w_k)(\mathbf{F}_k^T (x_k - \hat{x}_{k|k})^T + w_k^T) | Z^k] \\ \mathbf{P}_{k+1|k} &= E[\mathbf{F}_k (x_k - \hat{x}_{k|k})(x_k - \hat{x}_{k|k})^T \mathbf{F}_k^T + w_k w_k^T | Z^k] \\ \mathbf{P}_{k+1|k} &= \mathbf{F}_k E[(x_k - \hat{x}_{k|k})(x_k - \hat{x}_{k|k})^T | Z^k] \mathbf{F}_k^T + E[w_k w_k^T | Z^k] \\ \mathbf{P}_{k+1|k} &= E[(x_{k+1} - \hat{x}_{k+1|k})(x_{k+1} - \hat{x}_{k+1|k})^T | Z^k] = \mathbf{F}_k \mathbf{P}_{k|k} \mathbf{F}_k^T + \mathbf{Q}_k \end{aligned} \quad 2.10$$

Tendo obtido uma estimativa preditiva,  $\hat{x}_{k+1|k}$ , suponha que agora obtemos uma outra observação  $z_{k+1}$ . Podemos utilizá-la supondo que a nova estimativa (atualização) é uma soma linear ponderada da predição  $\hat{x}_{k+1|k}$  e da nova observação. Como é descrita na equação (2.11).

$$\hat{x}_{k+1|k+1} = \mathbf{K}'_{k+1} \hat{x}_{k+1|k} + \mathbf{K}_{k+1} z_{k+1} \quad 2.11$$

Onde os pesos, ou matrizes de ganho, são denotados por  $\mathbf{K}'_{k+1}$  e  $\mathbf{K}_{k+1}$ . Nosso problema é determinar  $\mathbf{K}'_{k+1}$  e  $\mathbf{K}_{k+1}$  de modo a minimizar o erro médio quadrático da estimação, onde o erro de estimação é dado por:

$$\tilde{x}_{k+1|k+1} = \hat{x}_{k+1|k+1} - x_{k+1} \quad 2.12$$

### 2.2.5. Condição de não tendenciosidade

Para que o filtro seja não tendencioso é necessário que  $E[\hat{x}_{k+1|k+1}] = E[x_{k+1}]$ . Supomos que  $\hat{x}_{k|k}$  é uma estimativa não tendenciosa. Substituindo a equação (2.2) na equação (2.11) e aplicando o operador esperança, temos:

$$\begin{aligned} E[\hat{x}_{k+1|k+1}] &= E[K'_{k+1}\hat{x}_{k+1|k} + K_{k+1}(H_{k+1}x_{k+1} + v_{k+1})] \\ E[\hat{x}_{k+1|k+1}] &= K'_{k+1}E[\hat{x}_{k+1|k}] + K_{k+1}H_{k+1}E[x_{k+1}] + K_{k+1}E[v_{k+1}] \end{aligned} \quad 2.13$$

Como o valor médio do ruído é zero:  $E[\hat{x}_{k+1|k+1}] = K'_{k+1}E[\hat{x}_{k+1|k}] + K_{k+1}H_{k+1}E[x_{k+1}]$ .

Aplicando o operador de expectância na eq. (2.9) e observando a condição de não tendenciosidade, temos:

$$E[\hat{x}_{k+1|k}] = \mathbf{F}_k E[\hat{x}_{k|k}] + \mathbf{G}_k u_k = E[x_{k+1}] \quad 2.14$$

Combinando as equações (2.13) e (2.14) temos:

$$E[\hat{x}_{k+1|k+1}] = \mathbf{K}'_{k+1}E[x_{k+1}] + \mathbf{K}_{k+1}\mathbf{H}_{k+1}E[x_{k+1}]$$

$$E[\hat{x}_{k+1|k+1}] = (\mathbf{K}'_{k+1} + \mathbf{K}_{k+1}\mathbf{H}_{k+1})E[x_{k+1}]$$

Observe que a condição de não tendenciosidade exige que:

$$\mathbf{K}'_{k+1} + \mathbf{K}_{k+1}\mathbf{H}_{k+1} = \mathbf{I} \text{ ou}$$

$$\mathbf{K}'_{k+1} = \mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1} \quad 2.15$$

Aplicando este último resultado a equação (2.11), temos:

$$\hat{x}_{k+1|k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})\hat{x}_{k+1|k} + \mathbf{K}_{k+1}z_{k+1}$$

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + \mathbf{K}_{k+1}(z_{k+1} - \mathbf{H}_{k+1}\hat{x}_{k+1|k}) \quad 2.16$$

A constante  $\mathbf{K}$  é conhecida como **Ganho de Kalman**.

Note que o termo  $\mathbf{H}_{k+1}\hat{\mathbf{x}}_{k+1|k}$  pode ser interpretado como uma previsão da observação  $z_{k+1}$  (veja figura 1). Assim poderemos interpretar a equação (2.16) com a soma da previsão mais uma fração da diferença entre a observação e a sua predição.

### 2.2.6. Determinando a covariância do erro

Determinamos a covariância do erro da Predição pela equação (1.10). Faremos agora, a atualização do erro da covariância.

$$\mathbf{P}_{k+1|k+1} = E[(\tilde{\mathbf{x}}_{k+1|k+1})(\tilde{\mathbf{x}}_{k+1|k+1})^T | Z^k]$$

$$\mathbf{P}_{k+1|k+1} = E[(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k+1})(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k+1})^T | Z^k]$$

$$\mathbf{P}_{k+1|k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})E[\ddot{\mathbf{x}}_{k+1|k}\ddot{\mathbf{x}}_{k+1|k}^T](\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})^T + \mathbf{K}_{k+1}E[\mathbf{v}_{k+1}\mathbf{v}_{k+1}^T]\mathbf{K}_{k+1}^T + 2(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})E(\ddot{\mathbf{x}}_{k+1|k}\mathbf{v}_{k+1}^T)\mathbf{K}_{k+1}^T$$

Tomando:

$$E[\mathbf{v}_{k+1}\mathbf{v}_{k+1}^T] = \mathbf{R}_{k+1}$$

$$E[\ddot{\mathbf{x}}_{k+1|k}\ddot{\mathbf{x}}_{k+1|k}^T] = \mathbf{P}_{k+1|k}$$

$$E[\ddot{\mathbf{x}}_{k+1|k}\mathbf{v}_{k+1}^T] = 0$$

Obtivemos,

$$\mathbf{P}_{k+1|k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})\mathbf{P}_{k+1|k}(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})^T + \mathbf{K}_{k+1}\mathbf{R}_{k+1}\mathbf{K}_{k+1}^T \quad 2.17$$

Assim a estimativa atualizada da covariância é expressa em termos da predição da covariância  $\mathbf{P}_{k+1|k}$ , do ruído da observação  $\mathbf{R}_{k+1}$  e da atriz ganho Kalman  $\mathbf{K}_{k+1}$ .

### 2.2.7 Escolhendo o Ganho de Kalman

Nosso objetivo é agora minimizar o erro de média quadrática condicional da estimativa com respeito ao ganho de Kalman,  $\mathbf{K}$ .

$$L = \min_{\mathbf{K}_{k+1}} E[\tilde{\mathbf{x}}_{k+1|k+1}^T \tilde{\mathbf{x}}_{k+1|k+1} | \mathbf{Z}^{k+1}]$$

$$L = \min_{\mathbf{K}_{k+1}} \text{trace} \left( E[\tilde{\mathbf{x}}_{k+1|k+1}^T \tilde{\mathbf{x}}_{k+1|k+1} | \mathbf{Z}^{k+1}] \right)$$

$$L = \min_{\mathbf{K}_{k+1}} \text{trace} (\mathbf{P}_{k+1|k+1}) \quad 2.18$$

Para qualquer matriz  $\mathbf{A}$  e uma matriz simétrica  $\mathbf{B}$ :

$$\frac{\partial}{\partial \mathbf{A}} \left( \text{trace}(\mathbf{A} \mathbf{B} \mathbf{A}^T) \right) = 2 \mathbf{A} \mathbf{B}$$

(para isto, consideramos escrever a linha como  $\sum_i a_i^T \mathbf{B} \mathbf{a}_i$  onde  $a_i$  são as colunas de  $\mathbf{A}^T$ , e então diferenciamos com respeito à  $a_i$ ).

Combinando as equações (17) e (18) e diferenciando a respeito da matriz de ganho (usando a relação acima) e igualando-a à zero resulta:

$$\frac{\partial L}{\partial \mathbf{K}_{k+1}} = 2(\mathbf{I} - \mathbf{K}_{k+1} \mathbf{H}_{k+1}) \mathbf{P}_{k+1|k} \mathbf{H}_{k+1}^T + 2 \mathbf{K}_{k+1} \mathbf{R}_{k+1}$$

Re-organizando temos a seguinte matriz de ganho:

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1|k} \mathbf{H}_{k+1}^T \left[ \mathbf{H}_{k+1} \mathbf{P}_{k+1|k} \mathbf{H}_{k+1}^T + \mathbf{R}_{k+1} \right]^{-1} \quad 2.19$$

Junto com a equação 16, isto define a média-quadrática linear ótima do erro do estimador.

### 2.2.8 Resumo das Equações Chaves

**Previsão:** Também conhecido como tempo de atualização. Ele prediz o estado e a variância do instante  $k+1$  dependendo da informação do instante  $k$ .

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{F}_k \hat{\mathbf{x}}_{k|k} + \mathbf{G}_k \mathbf{u}_k \quad 2.20$$

$$\mathbf{P}_{k+1|k} = \mathbf{F}_k \hat{\mathbf{P}}_{k|k} \mathbf{F}_k^T + \mathbf{Q}_k \quad 2.21$$

**Atualização:** Também conhecido como medida da atualização. Ele atualiza o estado e a variância usando a combinação do estado de previsão e da observação  $z_{k+1}$ .

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1} \left[ \mathbf{z}_{k+1} - \mathbf{H}_{k+1} \hat{\mathbf{x}}_{k+1|k} \right] \quad 2.22$$

$$\mathbf{P}_{k+1|k+1} = (\mathbf{I} - \mathbf{K}_{k+1} \mathbf{H}_{k+1}) \hat{\mathbf{P}}_{k+1|k} (\mathbf{I} - \mathbf{K}_{k+1} \mathbf{H}_{k+1})^T + \mathbf{K}_{k+1} \mathbf{R}_{k+1} \mathbf{K}_{k+1}^T \quad 2.23$$

onde a matriz de ganho é determinada por,

$$\mathbf{K}_{k+1|k+1} = \mathbf{P}_{k+1|k} \mathbf{H}_{k+1}^T \left[ \mathbf{H}_{k+1} \mathbf{P}_{k+1|k} \mathbf{H}_{k+1}^T + \mathbf{R}_{k+1} \right]^{-1} \quad 2.24$$

Junto com as condições iniciais na estimativa e sua matriz de covariância do erro (equação 2.6), a matriz de ganho define a seqüência discreta no tempo, um algoritmo recursivo para determinar a estimativa de variância mínima linear conhecida como Filtro de Kalman.

### 2.2.9 Interpretando o Filtro de Kalman

Olharemos agora, com mais detalhes, para o algoritmo global do Filtro de Kalman. A Figura 02 resume as fases do algoritmo na forma de um diagrama de bloco.

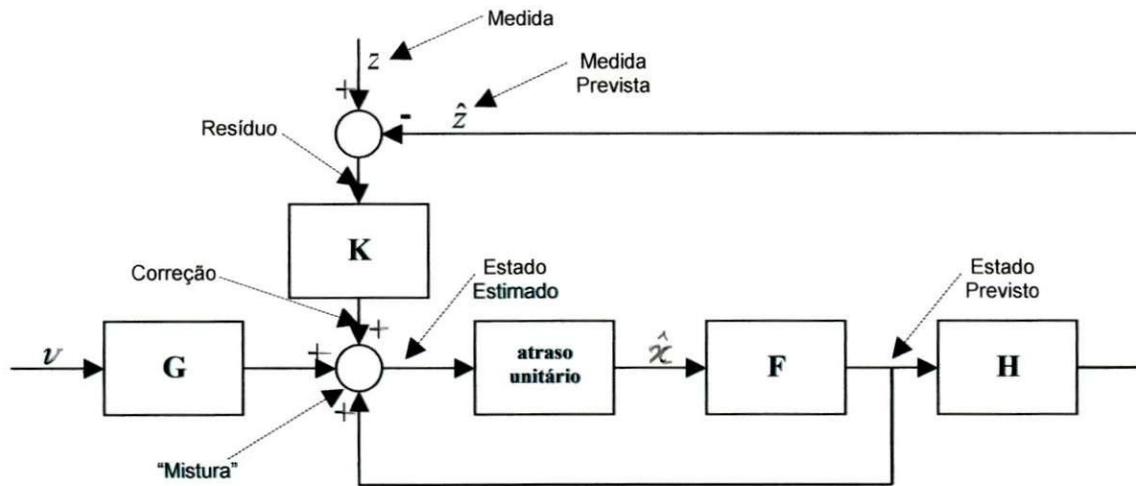


Figura 02 – Diagrama em Blocos do Filtro de Kalman Discreto no Tempo

A **inovação** (*innovation – whitening*),  $v_{k+1}$ , é definida como a diferença entre a observação (medida)  $z_{k+1}$  e a previsão  $\hat{z}_{k+1|k}$  feita com base na informação avaliada no instante  $k$ . Isto é uma medida da nova informação dada pela soma da outra medida no processo de estimação.

Dado que,

$$\begin{aligned} \hat{z}_{k+1|k} &= E[z_{k+1} | Z^k] \\ &= E[H_{k+1}x_{k+1} + v_{k+1} | Z^k] \\ &= H_{k+1}\hat{x}_{k+1|k} \end{aligned} \tag{2.25}$$

A **inovação**,  $v_{k+1}$  pode ser expressa por,

$$v_{k+1} = z_{k+1} - H_{k+1}\hat{x}_{k+1|k} \tag{2.26}$$

A inovação ou **resíduo** é uma medida importante da performance do estimador. Por exemplo, pode ser usado para validar uma medida antes dela ser incluída como parte da seqüência de observação (mais posteriormente).

O processo de transformação de  $\mathbf{z}_{k+1}$  em  $\mathbf{v}_{k+1}$  pode ser alcançado diretamente pelo **Filtro de Kalman Branco**. Isto se deve ao fato de que o *resíduo* forma uma seqüência não-correlacionada ortogonal do ruído branco do processo  $\mathbf{V}^{k+1}$  com o equivalente estatístico das observações  $\mathbf{Z}^{k+1}$ . Isto é importante porque como  $\mathbf{z}_{k+1}$  é geralmente estatisticamente correlacionada, o resíduo  $\mathbf{v}_{k+1}$  é não correlacionado, então, efetivamente isto nos dá uma nova informação ou “inovação” (que chamamos de resíduo).

O resíduo tem média zero, então:

$$\begin{aligned} E[\mathbf{v}_{k+1} | \mathbf{Z}^k] &= E[\mathbf{z}_{k+1} - \hat{\mathbf{z}}_{k+1|k} | \mathbf{Z}^k] \\ &= E[\mathbf{z}_{k+1} | \mathbf{Z}^k] - \hat{\mathbf{z}}_{k+1|k} \\ &= 0 \end{aligned} \quad 2.27$$

E a variância do Resíduo  $S_{k+1}$  é dado por:

$$\begin{aligned} S_{k+1} &= E[\mathbf{v}_{k+1} \mathbf{v}_{k+1}^T], \\ &= E\left[ \left( \mathbf{z}_{k+1} - \mathbf{H}_{k+1} \hat{\mathbf{x}}_{k+1|k} \right) \left( \mathbf{z}_{k+1} - \mathbf{H}_{k+1} \hat{\mathbf{x}}_{k+1|k} \right)^T \right] \\ S_{k+1} &= \mathbf{R}_{k+1} \mathbf{H}_{k+1} \mathbf{P}_{k+1|k} \mathbf{H}_{k+1}^T \end{aligned} \quad 2.28$$

Usando as equações 26 e 28 podemos reescrever a atualização de Kalman em termos do resíduo e da variância, como segue:

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k+1} &= \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1} \mathbf{v}_{k+1} \\ \mathbf{P}_{k+1|k+1} &= E\left[ \left( \mathbf{x}_{k+1} \hat{\mathbf{x}}_{k+1|k} - \mathbf{K}_{k+1} \mathbf{v}_{k+1} \right) \left( \mathbf{x}_{k+1} \hat{\mathbf{x}}_{k+1|k} - \mathbf{K}_{k+1} \mathbf{v}_{k+1} \right)^T \right] \end{aligned} \quad 2.29$$

$$\begin{aligned} \mathbf{P}_{k+1|k+1} &= E\left[ \left( \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k} \right) \left( \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k} \right)^T \right] - \mathbf{K}_{k+1} E[\mathbf{v}_{k+1} \mathbf{v}_{k+1}^T] \\ \mathbf{P}_{k+1|k+1} &= \mathbf{P}_{k+1|k} - \mathbf{K}_{k+1} \mathbf{S}_{k+1} \mathbf{K}_{k+1}^T \end{aligned} \quad 2.30$$

Onde, da equação 19:

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1|k} \mathbf{H}_{k+1}^T \mathbf{S}_{k+1}^{-1} \quad 2.31$$

e

$$\mathbf{S}_{k+1} = \mathbf{H}_{k+1} \mathbf{P}_{k+1|k} \mathbf{H}_{k+1}^T + \mathbf{R}_{k+1} \quad 2.32$$

Esta é uma forma conveniente do Filtro de Kalman, usada frequentemente em análises. Embora o algoritmo do Filtro de Kalman seja usado principalmente como um “estimador de



estados”, ele pode ser usado também para estimar outros parâmetros do vetor de estados. Ilustrados na Figura 2.

1. Se aplicado para estimar  $\hat{\mathbf{z}}_{k+1|k}$  é chamado de Filtro de Medida.
2. Se aplicado para estimar  $\hat{\mathbf{x}}_{k+1|k}$  é chamado de Filtro de Previsão.
3. Se aplicado para estimar  $\nu_{k+1}$  é chamado de Filtro de Branco.
4. Se aplicado para estimar  $\hat{\mathbf{x}}_{k+1|k+1}$  é chamado de Filtro de Kalman.

O Filtro de Kalman envolve uma seqüência de equações recursivas como foi determinado acima, para  $k=0,1,2,\dots$ , Logo abaixo temos um exemplo de um código de programa no Matlab que implementa o Filtro de Kalman para uma .

1. Ambos,  $K_{k+1}$  e  $P_{k+1|k+1}$  tendem para os valores constantes como  $k \rightarrow \infty$  (estado-fixo).
2. A estimativa  $\hat{\mathbf{x}}_{k+1|k+1}$  tende a seguir a medida avalia bastante de perto. Realmente desde que  $K$  é que uma função de pesagem que age na medida isto está clara que este efeito é mais proeminente quando  $K$  for alto.

```
% calcula as predicoes iniciais
function [xpred, Ppred] = predict(x, P, F, Q)
    xpred = F*x;
    Ppred = F*P*F' + Q;

% funcao que atualiza as predicoes
function [nu, S] = innovation(xpred, Ppred, z, H, R)

    nu = z - H*xpred;           %% atualização
    S = R + H*Ppred*H';        %% atualização de covariancia

%funcao [xnew, Pnew] = atualização do resíduo (xpred, Ppred, nu, S, H)
K = Ppred*H'*inv(S);          %% Ganho de Kalman
xnew = xpred + K*nu;          %% novo estado
Pnew = Ppred - K*S*K';       %% nova covariancia
```

Quadro 01 (a) – Código fonte do Matlab que implementa o Filtro de Kalman para representar a estimativa e observação de uma trajetória

```
%%% Documento Original do Matlab para simular dados e processos usando o
%%% Filtro de Kalman

delT = 1;

F = [ 1 delT                                %Matriz transição de estado
      0 1 ];
H = [ 1 0 ];                                %Matriz transição de saída.
Q = [ 1 1                                    %Matriz Covariância de ruído do processo
      1 1 ];
R = [ 1 ];                                   %Matriz Covariância de ruído da medição

z = [0.0 2.5 1 4 2.5 5.5 ]; %Vetor de observações no instante K
x = zeros(2,6); %Vetor estado do sistema no instante k.
P = zeros(2,2,6); %estimção da Covariância no instante k

x(:,1) = [ 0 ;10]; %Estimção de x no instante k baseada sobre
                  %o instante i, k>i
P(:,:,1) = [ 10 0; 0 10 ]; %Matriz Covariância
for i=2:6 % calculo das estimativas para o sistema
    [xpred, Ppred] = predict(x(:,i-1), P(:,:,i-1), F, Q);
    [nu, S] = innovation(xpred, Ppred, z(i), H, R);
    [x(:,i), P(:,:,i)] = innovation_update(xpred, Ppred, nu, S, H);
end
%Saida real e estimada do sistema
plot(z(2:6), 'b');
hold on;
plot(x(1,2:6), 'r');
axis([0 5 0 6]);
```

Quadro 01(b) – Código fonte do Matlab que implementa o Filtro de Kalman para representar a estimativa e observação de uma trajetória

O resultado do Programa apresentado no Quadro-01 pode ser visto na Figura 03. A observação e a estimativa seguem o mesmo padrão.

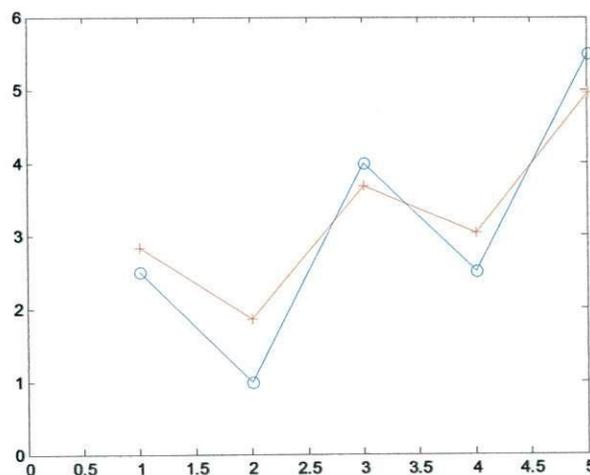


Figura 03 – Medidas (o) e estimativas (+) dos estados de uma trajetória.

## 2.3 Desempenho do Filtro de Kalman

Neste tópico consideramos como avaliar o desempenho de um Filtro de Kalman. Focamos-nos no entendimento dos seguintes problemas:

1. Como um Filtro de Kalman de um sistema perfeitamente modelado se comporta com o ruído perfeitamente estimado.
2. O efeito que as mudanças nos valores das fontes de ruído têm no desempenho global de um filtro Kalman.
3. Como reconhecer se as suposições do filtro estão de acordo com a prática. Isto é particularmente importante em muitas situações práticas desde que muitos sistemas reais não são bem representados por um modelo linear e a medida de ruído do sistema é não-Gaussiana.

Para o ponto (3), consideramos como descobrir um problema com um Filtro de Kalman. Consideramos também, como modificar o Filtro de Kalman para acomodar as não linearidades do processo, e, modelos de medida e ruídos não-Gaussianos na parte final.

### 2.3.1 Desempenho sob condições de modelagem ideal

Consideramos primeiro o desempenho do Filtro de Kalman sob condições de modelagem ideais significando que se conhece o modelo do sistema precisamente da forma dos modelos do processo e do ruído.

Na figura 04 mostramos a trajetória da posição alcançada aplicando o Filtro de Kalman. Na figura 04(a) mostramos as posições preditas e calculadas junto com as medidas para a trajetória completa. A trajetória inicial é mostrada de perto na Figura 04(b). A questão chave que notamos aqui é que a estimativa atualizada  $x_{k+1|k+1}$  sempre se situa entre a predição e a medida  $z_{k+1}$ . Isto segue do fato que a atualização é uma soma ponderada do prognóstico e da medida.

Na figura 04(c) mostramos de perto do estimador quando está no estado estacionário. Neste caso, os pesos (i.e. o ganho de Kalman) usados na atualização são aproximadamente constantes. Na figura 05 mostramos a trajetória da velocidade alcançada aplicando o Filtro de Kalman. Nenhuma medida é feita do estado de velocidade, assim são produzidas estimativas pela correlação cruzada entre a velocidade e posição (i.e. por  $P$ ).

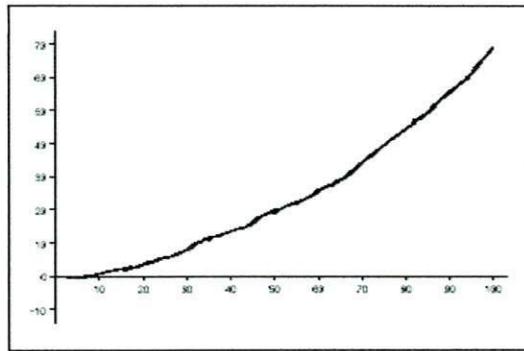


Figura 04: Trajetória da posição: a) Posições calculadas, preditas e observação;

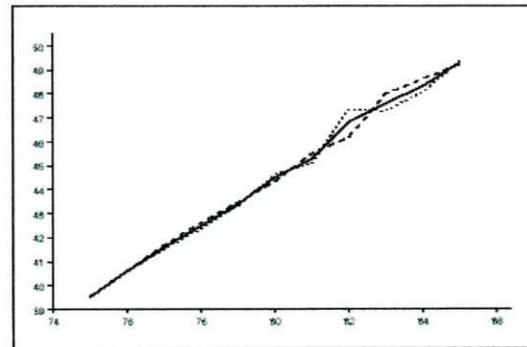
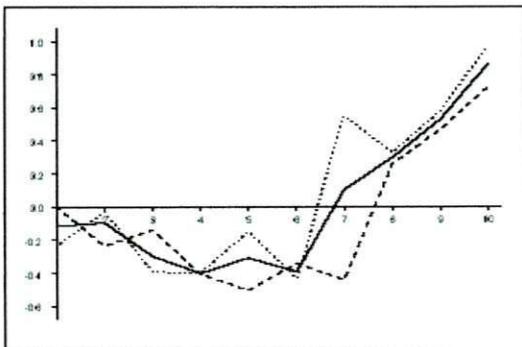


Figura 04: b) trajetória inicial; c) trajetória em estado estacionário.

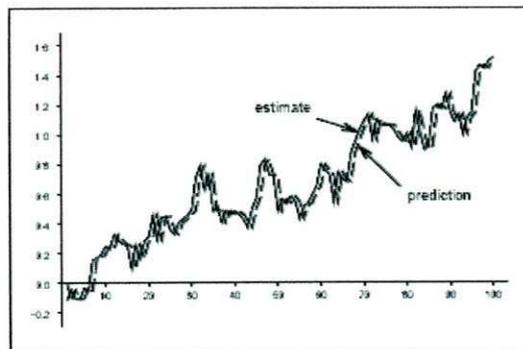


Figura 05: trajetória da velocidade

### 2.3.1.1 Desempenho do estado estacionário

Mostramos na figura 06 as covariâncias de erro predito e estimado para posição e velocidade. Em particular, note que eles tendem a valores constantes quando  $k$  é grande.

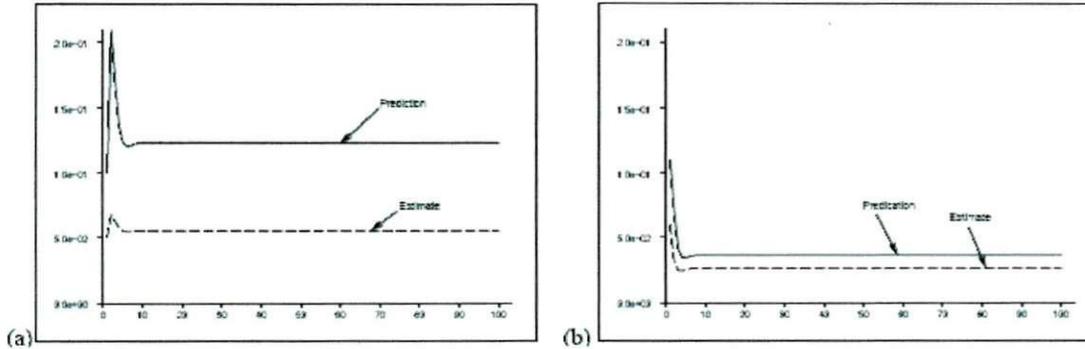


Figura 06: (a) covariância do erro de posição; (b) covariância do erro de velocidade.

O desempenho no estado estacionário depende dos valores escolhidos para o processo e as medidas das matrizes de covariância do ruído,  $\mathbf{Q}$  e  $\mathbf{R}$ .

Dado que

$$\mathbf{P}_{k+1|k} = \mathbf{F}_k \mathbf{P}_{k|k} \mathbf{F}_k^T + \mathbf{Q}_k$$

e

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{P}_{k|k-1} \mathbf{H}_k^T \left[ \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k \right]^{-1} \mathbf{H}_k \mathbf{P}_{k|k-1}$$

Temos

$$\mathbf{P}_{k+1|k} = \mathbf{F}_k \left[ \mathbf{P}_{k|k-1} - \mathbf{P}_{k|k-1} \mathbf{H}_k^T \left[ \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k \right]^{-1} \mathbf{H}_k \mathbf{P}_{k|k-1} \right] \mathbf{F}_k^T + \mathbf{Q}_k \quad 2.33$$

A equação 2.33 é conhecida como **Equação Matricial de Ricatti** discreta no tempo. Se, o sistema é invariante no tempo- (i.e.  $\mathbf{F}$ ,  $\mathbf{G}$  e  $\mathbf{H}$  são constantes), e a medida do ruído do processo são estacionários ( $\mathbf{Q}$  e  $\mathbf{R}$  são constantes) então como  $k \rightarrow \infty$  a solução para equação 2.33 converge para uma matriz definida positiva  $\bar{\mathbf{P}}$  contanto que o modelo de sistema seja completamente observável e completamente controlável. A matriz de ganho correspondente  $\bar{\mathbf{K}} = \bar{\mathbf{P}} \mathbf{H}^T \mathbf{S}^{-1}$  também será constante e chamada o ganho em estado estacionário.

A importância deste resultado é que em algumas aplicações você pode assumir que o Filtro de Kalman trabalha sob condições de estado estacionário. Neste caso você fixa o valor de  $\bar{\mathbf{P}}$  e conseqüentemente  $\bar{\mathbf{K}}$  desde o começo e condições iniciais não precisam ser especificadas. Uma vez que  $\mathbf{K}$  é agora fixo significa que podem ser feitas economias computacionais consideráveis desde que  $\mathbf{K}$  não tenha que ser recomputado a cada passo de tempo.

### 2.3.1.2 Inicialização

Recorde que parte das exigências para um Filtro de Kalman é a especificação das condições iniciais. Então, ao considerar a implementação de um Filtro de Kalman uma preocupação importante é como fixar (supor) valores para  $\mathbf{x}_{0|0}$  e  $\mathbf{P}_{0|0}$  já que eles não são conhecidos. A pergunta óbvia é o quanto sua suposição é boa (ou ruim)?

Uma possibilidade é inicializar o vetor de estado das medidas

$$\hat{\mathbf{x}}_{0|0} = \begin{bmatrix} z_0 \\ \frac{z_0 - z_{-1}}{\Delta t} \end{bmatrix}$$

e um modo simples para inicializar a matriz de covariância de estado é fixar isto para ser um múltiplo R da matriz de ruído do processo

$$\mathbf{P}_{0|0} = R\mathbf{Q}_k$$

Onde R é uma constante (tipicamente  $R = 10$ )

Na figura 07 ilustramos o efeito que a mudança dos parâmetros de inicialização tem sobre o desempenho do Filtro de Kalman por completo. Note que sem considerar os valores iniciais de  $\hat{\mathbf{x}}$  e  $\mathbf{P}$  eles tendem a valores constantes em poucas iterações.

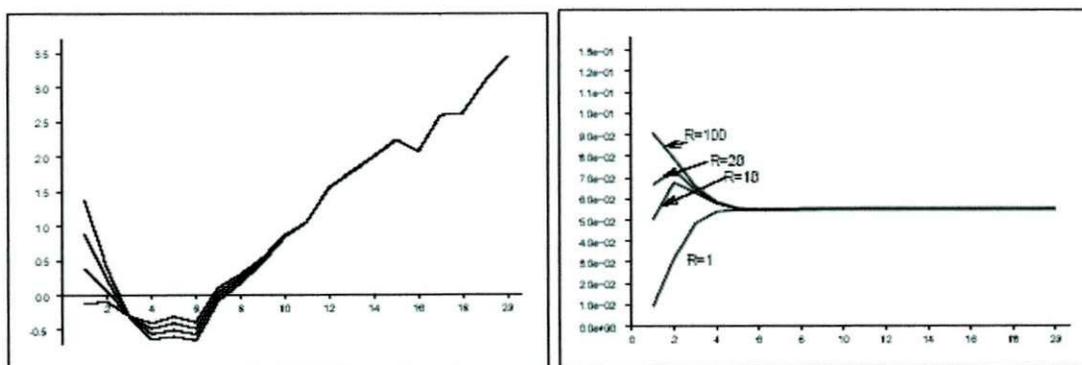


Figure 07: (a) Efeito de usar valores de inicialização de posição diferentes; (b) efeito de mudar R para inicializar  $\mathbf{P}_{0|0}$

Mais formalmente, pode ser mostrado que contanto que o sistema seja observável e controlável o erro devido a inicialização tende a zerar quando  $k \rightarrow \infty$ . Finalmente note que embora boa inicialização seja desejável para um Filtro de Kalman linear não é essencial (o estimador apenas leva muito mais tempo para se estabilizar). Porém, uma boa inicialização é crítica na implementação de filtros de Kalman para modelos de sistemas não-lineares.

### 2.3.1.3 Consistência de verificação

Desde que na prática não podemos medir o desempenho com respeito a medidas de erro de estado, uma vez que não sabemos os verdadeiros valores de estado, como conferimos que o filtro está executando corretamente? A resposta é que podemos definir medidas de desempenho do filtro em termos da **inovação (resíduo)**.

Sabemos que se o filtro está trabalhando corretamente então  $v_k$  tem média zero e é branco com uma covariância  $S_k$ . Assim podemos verificar que o filtro é consistente aplicando os dois procedimentos seguintes.

1. Cheque que o resíduo é consistente com sua covariância verificando que a magnitude do mesmo é limitada por  $\pm 2\sqrt{S_k}$ .
2. Verifique que o resíduo é imparcial e branco. Isto pode ser verificado usando teste de hipótese (teste  $\chi^2$ ).

**Teste 1 – teste do limite de magnitude do resíduo:** Na figura 08 mostramos o resíduo junto com os limites  $\pm\sigma$  e  $\pm 2\sigma$  em sua magnitude. A figura pareceria indicar que o resíduo é imparcial e aproximadamente 95% dos valores se situam dentro do limite como exigido. Este simples teste às vezes é suficiente para conferir a consistência do filtro. Porém, na prática é também mais habitual aplicar o próximo teste discutido.

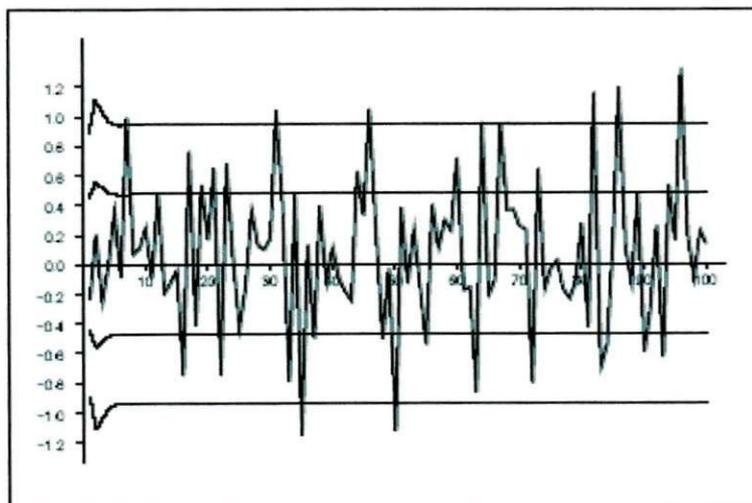


Figure 08: Resíduo e limites de desvio - padrão do resíduo.

**Teste 2 – Teste  $\chi^2$  normalizado quadrático dos resíduos:** para testar para imparcialidades computamos os resíduos normalizadas quadradas.

$$q_{k+1}(i) = v_{k+1}(i) S_{k+1}^{-1}(i) v_{k+1}(i) \quad 2.34$$

para uma seqüência de  $i$  tentativas de um Filtro de Kalman. Se as suposições de filtro são conhecidas então o  $q_{k+1}(i)$  são cada  $\chi^2$  em  $m$  graus de liberdade, onde  $m = 1$  em nosso caso (a dimensão do vetor de medidas). Assim,

$$E[q_{k+1}] = m \quad 2.35$$

Isto provê o teste para imparcialidade. Para estimar o significado precisamos ter  $N$  amostras independentes de  $q_{k+1}(i)$ ,  $i = 1, \dots, N$ . A média desta seqüência,

$$\bar{q}_{k+1} = \frac{1}{N} \sum_{i=1}^N q_{k+1}(i)$$

Que pode ser usada como uma estatística de teste uma vez que  $N\bar{q}_{k+1}$  é  $\chi^2$  com  $Nm$  graus de liberdade.

Em nosso caso, porém, podemos explorar o fato que o resíduo é ergódico<sup>3</sup> para calcular o significado da amostra da média de tempo para uma seqüência longa (isto é, a média do movimento) em lugar de uma média de conjunto. Assim, podemos calcular a média como,

$$\bar{q} = \frac{1}{N} \sum_{k=1}^N q_k \quad 2.36$$

de um único período do Filtro de Kalman. Com a Figura 09 mostramos o resíduo normalizado e a mudança média do mesmo. O posterior tende para 1.0 à medida que  $k$  cresce. Para testar a imparcialidade precisamos verificar que  $q$  se situa no intervalo de confiança  $[r_1, r_2]$  definido pela hipótese  $H_0$  que  $N\bar{q}$  é  $\chi^2_{Nm}$  distribuído com probabilidade  $1 - \alpha$ . Assim, precisamos encontrar  $[r_1, r_2]$  de tal forma que

$$P(N\bar{q} \in [r_1, r_2] | H_0) = 1 - \alpha \quad 2.37$$

<sup>3</sup> Hipótese ergódica. (Fís.) - Qualquer hipótese que permita provar a coincidência entre a média de uma variável de um sistema fechado tomada sobre o tempo e a média da mesma variável calculada no espaço de fase.

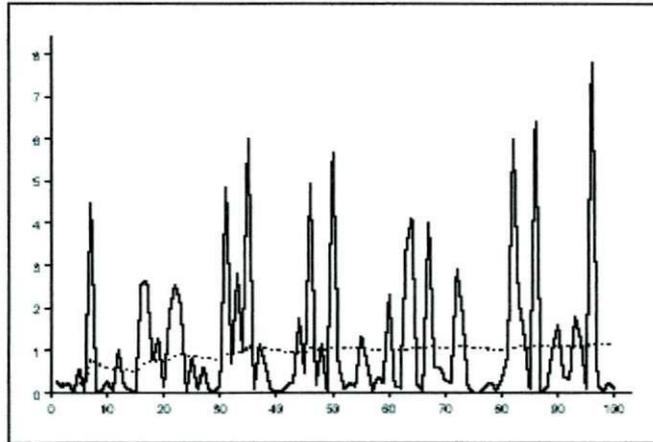


Figure 09: Resíduo normalizado e média de movimento

Para o exemplo, estamos considerando  $N = 100$ ,  $\bar{q} = 1.11$ , e faça  $\alpha = 0.05$  (isto é, definido para região de confiança 95% bi-lateral). Usando tabelas estatísticas achamos que,

$$[r_1, r_2] = [\chi_{100}^2(0.025), \chi_{100}^2(0.975)] = [74.22, 129.6]$$

A hipótese realmente é aceitável para este exemplo.

**Teste 3 - Teste do Resíduo Branco (auto-correlação):** Para testar a clareza precisamos provar que.

$$E[v_i^T v_j] = \mathbf{S}_i \delta_{ij} \quad 2.38$$

Podemos testar isto conferindo todos os pontos, exceto  $i = j$ , a estatística definida pela Equação 2.9 é zero dentro de erro estatístico permissível. Novamente, podemos explorar a ergodicidade para redefinir a estatística de teste como uma correlação tempo-médio.

$$r(\tau) = \frac{1}{N} \sum_{k=0}^{N-\tau-1} v_k^T v_{k+\tau} \quad 2.39$$

A auto-correlação é normalizada normalmente por  $r(0)$ .

Na Figura 10 mostramos a auto-correlação normalizada do resíduo para o exemplo que foi considerado. Note que os picos em  $\tau = 0$  e em todos os outros lugares é distribuído aleatoriamente em torno de zero. Podemos testar que as oscilações aproximadamente zero são aleatórias calculando a variância da estatística de teste. Para grandezas até  $N$  podemos assumir que  $r(\tau)$  é normalmente distribuído com zero médio e variância  $1/N$ . Então podemos computar

a janela  $2\sigma$  como  $\pm 2/\sqrt{N}$  e checar que em pelo, menos 95% dos valores caem dentro desta região de confiança. Novamente em nosso exemplo a auto-correlação satisfaz a hipótese.

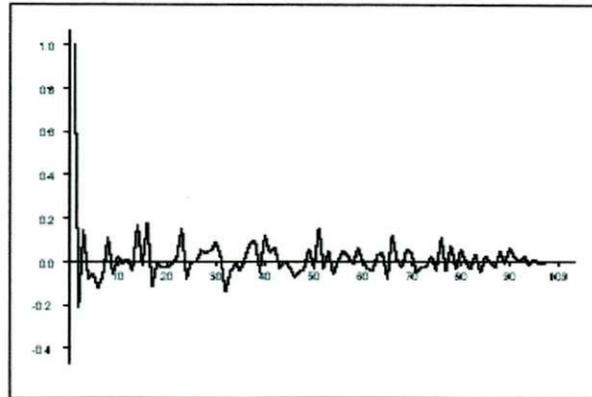


Figura 10: Auto-correlação para o resíduo

### 2.3.2 Validação do modelo

Até agora consideramos apenas o desempenho de um Filtro de Kalman quando o modelo do sistema e o ruído do processo forem precisamente conhecidos. Um Filtro de Kalman pode não executar corretamente se houver modelagem ou erro de estimativa de ruído ou ambos. Aqui discutimos as causas e identificamos a maioria das técnicas importantes usadas para controlar um Filtro de Kalman de divergência. Consideramos dois tipos de erro e as suas características;

1. Erro no processo e na especificação da observação do ruído.
2. Erro na modelagem dinâmica do sistema (modelo de processo).

#### Observações gerais:

- ✓ Se a relação do processo e ruído da medida é muito baixa, a sequência do resíduo se torna correlacionado.
- ✓ Os valores absolutos do processo e do ruído da medição podem ser fixados ajustando seus valores de forma que o teste do resíduo  $\chi^2$  seja satisfeito.
- ✓ No exemplo mostrado aqui, ajustar é muito mais sensível para às mudanças no ruído da medida que ao ruído processo. Neste exemplo, isto ocorre porque ruído da medida afeta a posição e o ruído do processo só afeta a velocidade.

### 2.3.2.1 Detectando erros de modelagem do processo

Consideramos agora o que acontece se tentarmos aplicar um estimador da medida de dados que não se ajustam ao modelo. O problema é denominado de *Filtro mal combinado*.

Especificamente, consideramos o caso de usar o Filtro de Kalman com uma velocidade constante para localizar uma partícula que tem uma verdadeira trajetória de movimento definida por um modelo de aceleração constante. Assim, o verdadeiro movimento é descrito pela equação de transição:

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{w}_k \quad 2.40$$

Escrevendo a matriz de transição de estados

$$\mathbf{F}_k = e^{\mathbf{A}\Delta t} = \begin{bmatrix} 1 & \Delta T & \Delta T^2 \\ 0 & 1 & \Delta T \\ 0 & 0 & 1 \end{bmatrix} \quad 2.41$$

Com

$$\mathbf{Q}_k = E[\mathbf{w}_k \mathbf{w}_k^T] = q \begin{bmatrix} \Delta T^5/20 & \Delta T^4/8 & \Delta T^3/6 \\ \Delta T^4/8 & \Delta T^3/3 & \Delta T^2/2 \\ \Delta T^3/6 & \Delta T^2/2 & \Delta T \end{bmatrix} \quad 2.42$$

## 2.4 Filtro de Kalman Estendido

Esta seção trata de uma variação do Filtro de Kalman discreto que é útil quando algumas das suposições da estrutura convencional do Filtro de Kalman não funcionam. Recordemos que os três pontos chaves do problema para um Filtro de Kalman são:

1. **Inicialização:** Supomos que o vetor do estado inicial e sua matriz de covariância do erro são conhecidos.
2. **Modelagem:** Supomos que temos um modelo linear exato do sistema do processo e da medida.
3. **Ruído:** Supomos que o ruído do processo e das medições são Gaussianos.

Nesta seção veremos como tratar de cada um destes problemas.



O Filtro de Kalman estendido é aplicado à modelos de sistema não lineares. Supomos que o sistema pode ser representado por um modelo não-linear do de espaço de estado discreto no tempo.

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, k) + \mathbf{w}_k \quad 2.43$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k, k) + \mathbf{v}_k \quad 2.44$$

Onde  $\mathbf{f}(\cdot, \cdot, k)$  é uma matriz não-linear da transição de estado e  $\mathbf{h}(\cdot, k)$  é uma matriz não linear da observação. Supomos que os ruídos do processo e da medição são Gaussianos, não correlacionados e com zeros não significativos, assim:

$$\begin{aligned} E[\mathbf{w}_k] &= 0 \\ E[\mathbf{v}_k] &= 0 \\ E[\mathbf{w}_i \mathbf{w}_j^T] &= \delta_{ij} \mathbf{Q}_i \\ E[\mathbf{v}_i \mathbf{v}_j^T] &= \delta_{ij} \mathbf{R}_i \\ E[\mathbf{w}_i \mathbf{v}_j^T] &= 0 \end{aligned}$$

### 2.4.1 Predição

Como no caso linear, supomos que temos no tempo  $k$

$$\hat{\mathbf{x}}_{k|k} = E[\mathbf{x}_k | \mathbf{Z}^k], \quad \mathbf{P}_{k|k}$$

Para gerar a predição expandimos a equação 4.10 em séries de Taylor sobre a predição  $\hat{\mathbf{x}}_{k|k}$  em termos de primeira ordem:

$$\mathbf{x}_{k+1} = \mathbf{f}(\hat{\mathbf{x}}_{k|k}, \mathbf{u}_k, k) + \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] [\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}] + \mathcal{Q}([\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}]^2) + \mathbf{w}_k \quad 2.45$$

Assim o jacobiano de  $\mathbf{f}$  é avaliado para  $\hat{\mathbf{x}}_{k|k}$ . Fazendo os cálculos da equação 2.45 e ignorando requisitos dos primeiros termos, podemos supor  $\hat{\mathbf{x}}_{k|k}$  é aproximadamente igual a média condicional do o ruído do processo.

$$\begin{aligned} \hat{\mathbf{x}}_{k|k} &= E[\mathbf{x}_k | \mathbf{Z}^k] \\ &= \mathbf{f}(\hat{\mathbf{x}}_{k|k}, \mathbf{u}_k, k) \end{aligned} \quad 2.46$$

A covariância do estado pode ser encontrada como segue. O erro da predição é dado primeiramente por:

$$\begin{aligned}
 \tilde{\mathbf{x}}_{k+1|k} &= \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k} \\
 &= \mathbf{f}(\hat{\mathbf{x}}_{k|k}, \mathbf{u}_k, k) + \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] [\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}] + \mathbf{Q} \left( [\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}]^2 \right) + \mathbf{w}_k - \mathbf{f}(\hat{\mathbf{x}}_{k|k}, \mathbf{u}_k, k) \\
 &\approx \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] [\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}] + \mathbf{w}_k \\
 \tilde{\mathbf{x}}_{k+1|k} &= \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] [\tilde{\mathbf{x}}_{k|k}] + \mathbf{w}_k
 \end{aligned} \tag{2.47}$$

A covariância da predição é encontrado então fazendo exame da expectativa do produto do erro da predição:

$$\begin{aligned}
 \mathbf{P}_{k+1|k} &= E \left[ \tilde{\mathbf{x}}_{k+1|k} \tilde{\mathbf{x}}_{k+1|k}^T \mid \mathbf{Z}^k \right] \\
 &\approx E \left[ \left( \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] [\tilde{\mathbf{x}}_{k|k}] + \mathbf{w}_k \right) \left( \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] [\tilde{\mathbf{x}}_{k|k}] + \mathbf{w}_k \right)^T \mid \mathbf{Z}^k \right] \\
 &= \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] E \left[ \tilde{\mathbf{x}}_{k|k} \tilde{\mathbf{x}}_{k|k}^T \mid \mathbf{Z}^k \right] \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right]^T + E \left[ \mathbf{w}_k \mathbf{w}_k^T \right] \\
 \mathbf{P}_{k+1|k} &= \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] \mathbf{P}_{k|k} \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right]^T + \mathbf{Q}_k
 \end{aligned} \tag{2.48}$$

Note que a covariância da predição tem a mesma que seu equivalente linear com o Jacobiano com o papel de matriz de transição  $\mathbf{F}_k$

### 2.4.2 Predição e resíduo da observação

A observação  $\mathbf{z}$  pode ser escrito como uma série de Taylor expandida sobre a predição

$\hat{\mathbf{x}}_{k+1|k}$ .

$$\mathbf{z}_{k+1} = \mathbf{h}(\hat{\mathbf{x}}_{k+1|k}) + \left[ \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right] [\hat{\mathbf{x}}_{k+1|k} - \mathbf{x}_{k+1}] + \mathbf{Q} \left( [\hat{\mathbf{x}}_{k+1|k} - \mathbf{x}_{k+1}]^2 \right) + \mathbf{w}_{k+1}$$

Truncando os termos de primeira ordem e fazendo um exame de expectativas sobre as observações

$$\hat{\mathbf{z}}_{k+1|k} \approx \mathbf{h}(\hat{\mathbf{x}}_{k+1|k}) \tag{2.49}$$

O resíduo é encontrado então como

$$\mathbf{v}_{k+1} = \mathbf{z}_{k+1} - \mathbf{h}(\hat{\mathbf{x}}_{k+1|k}) \quad 2.50$$

E a covariância do resíduo é encontrada como segue:

$$\begin{aligned} \mathbf{S}_{k+1} &= E[\mathbf{v}_{k+1}\mathbf{v}_{k+1}^T] \\ &= E\left[\left(\mathbf{z}_{k+1} - \mathbf{h}(\hat{\mathbf{x}}_{k+1|k})\right)\left(\mathbf{z}_{k+1} - \mathbf{h}(\hat{\mathbf{x}}_{k+1|k})\right)^T\right] \\ &\approx E\left[\left(\left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right](\mathbf{x}_{k+1|k} - \mathbf{x}_{k+1}) + \mathbf{w}_{k+1}\right)\left(\left(\mathbf{x}_{k+1|k} - \mathbf{x}_{k+1}\right)^T \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right]^T + \mathbf{w}_{k+1}^T\right)\right] \\ \mathbf{S}_{k+1} &= \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right] \mathbf{P}_{k+1|k} \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right]^T + \mathbf{R}_{k+1} \end{aligned} \quad 2.51$$

Onde o jacobiano  $\mathbf{h}$  é avaliado para  $\hat{\mathbf{x}}_{k+1|k}$ . Note outra vez que a equação 4.18 está no mesmo formato que suas contrapartes lineares exceto por  $\left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right]$  que substituiu  $\mathbf{H}_{k+1}$ .

### 2.4.3 Atualização

Pelo raciocínio similar ao feito para o caso linear é possível derivar dos primeiros princípios as equações para o ganho do filtro, a atualização do estado e o atualização da covariância.

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1|k} \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right]^T \mathbf{S}_{k+1}^{-1} \quad 2.52$$

A atualização do estado é dado por

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1} \left[\mathbf{z}_{k+1} - \mathbf{h}(\hat{\mathbf{x}}_{k+1|k})\right] \quad 2.53$$

E A atualização da covariância é dado por

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{K}_{k+1} \mathbf{S}_{k+1} \mathbf{K}_{k+1}^T \quad 2.54$$



### 2.4.4 Resumo das Equações

#### Predição:

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{f}(\hat{\mathbf{x}}_{k|k}, \mathbf{u}_k, k) \quad 2.55$$

$$\mathbf{P}_{k+1|k} = \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] \mathbf{P}_{k|k} \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right]^T + \mathbf{Q}_k \quad 2.56$$

#### Atualização:

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1} \left[ \mathbf{z}_{k+1} - \mathbf{h}(\hat{\mathbf{x}}_{k+1|k}) \right] \quad 2.57$$

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{K}_{k+1} \mathbf{S}_{k+1} \mathbf{K}_{k+1}^T \quad 2.58$$

Onde,

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1|k} \left[ \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right]^T \mathbf{S}_{k+1}^{-1} \quad 2.59$$

e,

$$\mathbf{S}_{k+1} = \left[ \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right] \mathbf{P}_{k+1|k} \left[ \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right]^T + \mathbf{R}_{k+1} \quad 2.60$$

#### Alguns comentários gerais

1. Os jacobianos  $\left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right]$  e  $\left[ \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right]$  são as funções do estado e do tempo; não são constantes;
2. **Estabilidade:** Desde que estamos tratando dos modelos de perturbação das matrizes do estado e da observação sobre a predição da trajetória, é importante que as predições sejam próximas dos estados verdadeiros, pois do contrário o filtro possivelmente divergirá.
3. **Iniciação:** Ao contrário do caso linear, temos que ter um cuidado especial ao inicializar o Filtro de Kalman Estendido.
4. **Peso computacional:** O Filtro de Kalman Estendido é computacionalmente mais significativamente pesado que os filtros lineares. Isto limitou seu uso nas aplicações. Entretanto, hoje as execuções computando em tempo real do EKF podem ser conseguidas usando recursos computacionais moderados.



### 2.4.5 Divergência no Filtro

Os ruídos tem uma importância fundamental no que tange ao comportamento dos processos de estimação. Inúmeros trabalhos tem sido desenvolvidos de maneira a elucidar a influência dos ruídos no processo de estimação do filtro.

Ressaltamos aqui um dos problemas principais abordados nestes trabalhos: a **divergência**. Diz-se que ocorre divergência quando, após um certo período de operação, os erros na estimativa eventualmente divergem dos valores reais preditos pelas equações do filtro. A explicação mais freqüente para este fenômeno é que as covariâncias calculadas dos erros nas estimativas podem tornar-se muito pequenas. Como resultado a predição através do modelo torna-se mais importante na estimativa do que as medidas subseqüentes, causando uma degradação significativa na precisão. Este efeito pode ser devido a uma infinidade de causas; em geral a imprecisão na modelagem ou mesmo o procedimento computacional adotado podem fazer com que o cálculo da matriz de covariância se torne um tanto quanto irreal.

A escolha de um nível apropriado para os elementos da matriz dos ruídos de entrada do sistema é heurística, dependendo em grande parte do conhecimento sobre as acelerações não modeladas.

Dada a não linearidade do problema em questão, tornou-se necessário o recurso a um Filtro de Kalman Estendido. Não há, portanto, garantias teóricas de otimização e nem de convergência deste método. Por isso, procuramos implementar um modelo de simulação que permite, debaixo de condições próximas da realidade verificar o desempenho desta técnica. O método apresentado não faz uma localização absoluta instantânea mas medidas sucessivas levam a que o estado estimado convirja para o estado real do robô.

### 3 Implementação e Resultados

A questão da implementação é similar a do Filtro linear de Kalman e você pode testar o desempenho do filtro usando todas as técnicas apresentadas na seção 2.

Em particular, temos que ter o cuidado de fazer um estudo e verificar se o processo do sistema e do ruído estão modelados corretamente. Há obviamente alguns erros introduzidos usando a linearização do modelo.

Um ponto importante adicional para notar é que a matriz de covariância do estado é só uma aproximação da média quadrática do erro e não uma covariância verdadeira. Lembrando que  $\mathbf{P}_{k+1|k+1}$  determina o peso dado a medidas novas no procedimento de atualização. Assim, se  $\mathbf{P}_{k+1|k+1}$  é escolhido de modo errado e fica pequeno, as medidas são pequenas e afetam na estimação o que provavelmente fará com que o EKF venha a divergir.

#### 3.1 Modelo do Robô

Vamos considerar um robô com tração diferencial e que não há escorregamento de qualquer tipo, nomeadamente lateral, ou seja, que a velocidade das rodas, no ponto de contacto com o chão é sempre perpendicular ao seu eixo.

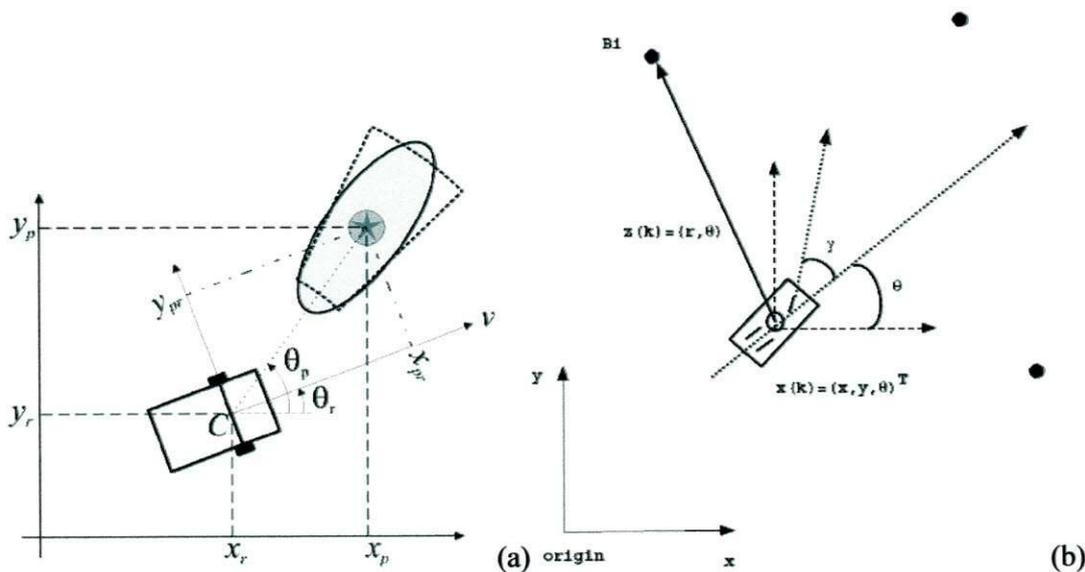


Figura 11: (a) Representação gráfica do estado associado ao Robô. (b) Geometria de um veículo-robô, explicitando o observador.

Considere a descrição do movimento de um veículo robótico ilustrado na Figura 11. Conseqüentemente, o estado do Robô, em termos de posicionamento e velocidade, pode ser representado por:

$$X(t) = [x(t) \ y(t) \ \theta(t) \ v(t) \ \omega(t)]^T \quad 3.1$$

Em que  $x(t)$  e  $y(t)$  representam a posição do ponto C no plano (Figura 11(a)),  $\theta(t)$  a atitude do Robô,  $v(t)$  a velocidade tangencial do ponto C e  $\omega(t)$  a velocidade angular, isto é, a velocidade de rotação do Robô segundo o eixo vertical que passa por C. Entretanto, queremos à priori controlar a trajetória, então, separando a parte cinemática de fora, podemos reescrever a equação 3.1 para o domínio discreto. O estado simplificado é agora descrito no tempo discreto como  $\mathbf{x}_k = [x_k, y_k, \phi_k]^T$ . O controle é provido por um vetor de controle de entrada que determina a velocidade da viagem  $\mathbf{u}_k = [V_k, \varphi_k]^T$ , então nosso sistema será:

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \phi_{k+1} \end{bmatrix} = \begin{bmatrix} x_k + \Delta t V_k \cos(\phi_k + \varphi_k) \\ y_k + \Delta t V_k \sin(\phi_k + \varphi_k) \\ \phi_k + \frac{V_k}{B} \Delta t \sin(\varphi_k) \end{bmatrix} + \mathbf{q}_k \quad 3.2$$

Onde,  $\Delta t$  é o intervalo de tempo entre as medidas.  $B$  é a linha de base da roda,  $\mathbf{q}_k$  é o vetor do ruído que combina erros de modelagem do processo e do controle.

Supomos que medidas da escala (profundidade) e o rolamento é feito a um jogo das localizações fixas em  $\mathbf{B}_i = [X_i, Y_i]^T$ ,  $i = 1, \dots, N$ . O modelo não-linear da observação é conseqüentemente:

$$\mathbf{z}_k = \begin{bmatrix} r_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} \sqrt{(X - x_k)^2 + (Y - y_k)^2} \\ \tan^{-1}\left(\frac{Y - y_k}{X - x_k}\right) - \phi_k \end{bmatrix} + \mathbf{r}_k \quad 3.3$$

Onde  $\mathbf{r}_k$  é o erro na medição.

## Predição

Da equação 2.46 o estado de predição  $\hat{\mathbf{x}}_{k+1|k}$  é dado por:

$$\begin{bmatrix} \hat{x}_{k+1|k} \\ \hat{y}_{k+1|k} \\ \hat{\phi}_{k+1|k} \end{bmatrix} = \begin{bmatrix} \hat{x}_{k|k} + \Delta t V_k \cos(\hat{\phi}_{k|k} + \varphi_k) \\ \hat{y}_{k|k} + \Delta t V_k \sin(\hat{\phi}_{k|k} + \varphi_k) \\ \hat{\phi}_{k|k} + \frac{V_k}{B} \Delta t \sin(\varphi_{k|k}) \end{bmatrix} + \mathbf{q}_k \quad 3.2$$

A predição da matriz de covariância é:

$$\mathbf{P}_{k+1|k} = \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] \mathbf{P}_{k|k} \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right]^T + \mathbf{Q}_k \quad 3.3$$

Onde,

$$\left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] = \begin{bmatrix} 1 & 0 & -\Delta t V_k \sin(\hat{\phi}_{k|k} + \varphi_k) \\ 0 & 1 & +\Delta t V_k \cos(\hat{\phi}_{k|k} + \varphi_k) \\ 0 & 0 & 1 \end{bmatrix} \quad 3.4$$

## Atualização:

As equações para atualizar o estado da covariância são:

$$\begin{cases} \hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1} [\mathbf{z}_k - \mathbf{h}(\hat{\mathbf{x}}_{k+1|k})] \\ \mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{K}_{k+1} \mathbf{S}_{k+1} \mathbf{K}_{k+1}^T \end{cases} \quad 3.5$$

Onde,

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1|k} \left[ \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right]^T \mathbf{S}_{k+1}^{-1} \quad 3.6$$

e,

$$\mathbf{S}_{k+1} = \left[ \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right] \mathbf{P}_{k+1|k} \left[ \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right]^T + \mathbf{R}_{k+1} \quad 3.7$$

Onde também,

$$\left[ \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right] = \begin{bmatrix} (\hat{x}_{k+1|k} - X)/d & (\hat{y}_{k+1|k} - Y)/d & 0 \\ -(\hat{y}_{k+1|k} - Y)/d^2 & (\hat{x}_{k+1|k} - X)/d^2 & -1 \end{bmatrix} \quad 3.8$$

Onde  $d = \sqrt{(X - \hat{x}_{k+1|k})^2 + (Y - \hat{y}_{k+1|k})^2}$ .

### 3.2 Implementação do Filtro de Kalman no Matlab® para as Trajetórias Circular e Elíptica

Baseado nas equações do Filtro de Kalman Estendido apresentadas na seção anterior, escolhemos dois tipos de trajetória para nosso Robô, a primeira, uma trajetória circular; a outra, uma generalização da primeira, uma trajetória elíptica. A trajetória circular foi escolhida por ser simples e abrir caminho para a implementação da elíptica, que tem bastante uso na área da robótica, um exemplo é o controle de satélites em órbita, que geralmente cumprem uma trajetória elíptica, e, em muitos casos o controle é feito através do Filtro de Kalman. Tomamos como base para descrever as trajetórias do robô as equações paramétricas da circunferência.

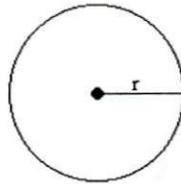


Figura 12 – circunferência.

Definida por:

$$x^2 + y^2 = r^2$$

onde,  $x = r \cos \varphi$   
 $y = r \sin \varphi$

Podemos obter a elipse partindo de suas equações paramétricas:

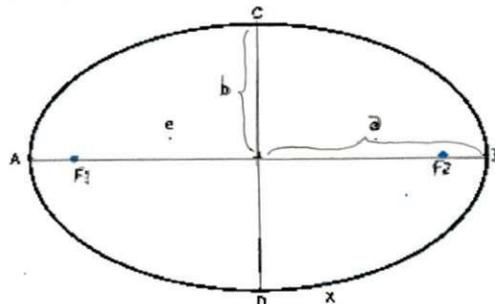


Figura 13 – Elipse

Definida por:

$$\left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 = 1 - \text{Elipse centrada na origem. Definimos sua equações paramétricas como:}$$

$$\begin{cases} x = a \cos \varphi \\ y = b \sin \varphi \\ r = \sqrt{a^2 \cos^2 \theta + b^2 \sin^2 \theta} \end{cases} \quad \text{onde, } a \text{ e } b \text{ são os semi-eixos da elipse.}$$



Ou ainda, “achatando” um lado da circunferência, que neste caso, é o mais interessante à fazer, já que já dispomos da implementação feita para a circunferência.

No quadro abaixo apresentamos a rotina do Matlab<sup>®</sup> que implementa um filtro de Kalman para o controle de uma trajetória circular de um robô.

```
% nome do arquivo twodim_nonlin: sistema modelado por tres graus de liberdade
clear all;
close all;
clc;

% definição de parâmetros
% intervalo entre os passos
delT = 1;
% vetor inicial de estado do sistema
x = [ 0; -2; 0 ];
% vetor velocidade de controle
u = [ -1; 0.1 ];
% posição do sinalizador
XB = [5, 5];

% matriz de covariancia do ruído no sistema
Q = [0.1 0 0
     0 0.1 0
     0 0 0.0002];
% matriz de covariancia do ruído na medição
R = [ 0.01 0
     0 0.00001 ];
%% Posicao inicial
xhat = [0; -2; 0 ];

% matriz de covariancia da previsão
P = 10*Q;

% Geração do Estado "Real"
for i=1:32
    w = gennormal([0;0;0], Q);           % ruído gaussiano
    x = robot(x, u) + 0.01*w;          % atualizacoes, modelo do sistema

    x(1:2)
    plot(x(1),x(2),'*k');
    plotrobot(x,'b');
    hold on;
    plot(XB(1),XB(2),'ok');
    axis([-10 10 -10 10]);
    pause(1);

% GERAÇÃO DAS MEDIDAS
    v = gennormal([0;0], R);
    z = beacon(x,XB) + v;              % simulação das medições
% observe que o x é obtido do modelo do sistema
% PREVISAO
    [xpred, Ppred] = predict_nonlin('robot', xhat, u, P, Q);
    plot(xpred(1), xpred(2),'+k');
    plot_covar(xpred(1:2),Ppred(1:2,1:2),'g');
    pause(1);
% ATUALIZAÇÃO
    [xhat, P] = update_nonlin(xpred, Ppred, z, 'beacon', XB, R);
    plot(xhat(1),xhat(2),'*k');
    plot_covar(xhat(1:2),P(1:2,1:2),'r');
    pause(1);
end
```

Quadro 02 - Filtro de Kalman para controle de um robô em uma trajetória circular.<sup>4</sup>

<sup>4</sup> Ver no Quadro 01: implementação das funções: *predict*, *innovation*.

```
%%% função robot - Geração de uma matriz de estado estimado, xnew exibe informações
function [xnew, xdf] = robot(x, u)
B = 0.5; % largura do pneu
%%%
xnew(1) = x(1) + u(1)*cos(x(3)+u(2));
xnew(2) = x(2) + u(1)*sin(x(3)+u(2));
xnew(3) = x(3) + u(1)/B*sin(u(2));
%%%
xdf = [1, 0, -u(1)*sin(x(3)+u(2)),
       0, 1, u(1)*cos(x(3)+u(2)),
       0, 0, 1];
```

Quadro 03 – Código que gera o estado do robô realizando uma trajetória circular.

O código do quadro 02 foi usado para simular o comportamento (estado) de um robô em três circunstâncias: estado esperado (desejado), estado medido (observado) e estado previsto (estimação através do Filtro de Kalman), quando o robô exerce uma trajetória circular. O modelo acrescenta ruído gaussiano ao sistema medido para fazê-lo mais próximo do que acontece realmente, além do ruído na previsão. Lembrando que a avaliação do filtro é feita através do resíduo. E no quadro-03 geramos a matriz de estados do robô, que define sua trajetória, neste caso circular.

A figura 14 foi gerada pelo Matlab<sup>®</sup> através do código apresentado no quadro-02, representamos as posições: *desejada* (simbolicamente representada por - o) junto com um retângulo que representa nosso robô; *observada* (simbolicamente representada por - +) e *prevista pelo Filtro de Kalman* (simbolicamente representada por - \*), ambas com elipses à volta representando sua covariância.

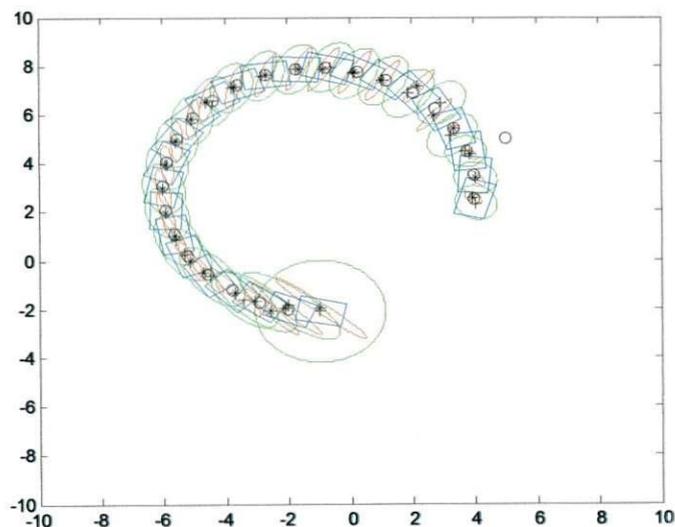


Figura 14 - Trajetória circular: Desejada (o), Observada (+) e Prevista pelo Filtro de Kalman (\*), cortada antes de divergir.



No quadro abaixo apresentamos a rotina do Matlab<sup>®</sup> que implementa um filtro de Kalman para o controle de uma trajetória elíptica de um robô, baseada na rotina anterior, para a trajetória circular e evidenciando neste modelo o estado desejado, observado e previsto.

```
% nome do arquivo twodim_nonlin
% sistema modelado por tres graus de liberdade
clear all;
close all;
clc;

% definição de parâmetros
% intervalo entre os passos
deltT = 1;
% vetor inicial de estado do sistema:
% gera uma matriz coluna vetor estado do sistema no instante k.alterado de 2 p 3
x = [ 0; -3; 0 ];
% vetor velocidade de controle
u = [ -1; 0.1 ]; % gera uma matriz coluna
XB = [7, 7]; % posição do observador alterada de 5 para 7
% matriz de covariancia do ruído no sistema/processo
Q = [0.1 0 0
     0 0.1 0
     0 0 0.0002];
% matriz de covarinancia do ruído na medição
R = [ 0.01 0
     0 0.00001 ];
%% Posicao inicial
xhat = [0; -2; 0 ];

% matriz de covariancia da previsão
P = 10*Q;
% Geração do Estado "Real"
for i=1:30 % Numero de iterações, usado também com 23.
    w = gennormal([0;0;0], Q); % geração de ruído gaussiano no processo
    x = robot(x, u); %+ w; % atualização dos estados
    x(1:2)
    plot(x(1),x(2),'ok'); %
    plotrobot(x,'b');
    hold on;
    plot(XB(1),XB(2),'ro');
    axis([-12 12 -12 12]);
    pause(1);

% GERAÇÃO DAS MEDIDAS
v = gennormal([0;0], R);
z = beacon(x,XB) + v; % simulação das medições
% observe que o x é obtido do modelo do sistema

% PREVISAO
[xpred, Ppred] = predict_nonlin('robot', xhat, u, P, Q);
plot(xpred(1), xpred(2),'+k');
plot_covar(xpred(1:2),Ppred(1:2,1:2),'g');
pause(1);

% ATUALIZAÇÃO
[xhat, P] = update_nonlin(xpred, Ppred, z, 'beacon', XB, R);
plot(xhat(1),xhat(2),'*k');
plot_covar(xhat(1:2),P(1:2,1:2),'r');
pause(1);
end
```

Quadro 04 – Filtro de Kalman para controle de um robô em uma trajetória elíptica.<sup>5</sup>

<sup>5</sup> Ver no Quadro 01: implementação das funções: *predict, innovation*.

```
%%% função robot - Geração de uma matriz de estado estimado, xnew exibe informações
function [xnew, xdf] = robot(x, u)
B = 0.5; % largura do pneu
%%% Criamos uma Elipse alongando a circunferencia em 65% no eixo-x
xnew(1) = x(1) + u(1)*1.65*cos(x(3)+u(2));
xnew(2) = x(2) + u(1)*sin(x(3)+u(2));
xnew(3) = x(3) + u(1)/B*sin(u(2));

xnewdot = [ 1 0 -u(1)*sin(x(3)+u(2))
            0 1 u(1)*cos(x(3)+u(2))
            0 0 1 ];
```

Quadro 05 – Código que gera o estado do robô realizando uma trajetória elíptica.

O código do quadro 04 foi usado para simular o comportamento (estado) de um robô em três circunstâncias: estado esperado (desejado), estado medido (observado) e estado previsto (estimação através do Filtro de Kalman), quando o robô exerce uma trajetória circular. O modelo acrescenta ruído gaussiano ao sistema medido para fazê-lo mais próximo do que acontece realmente, além do ruído na previsão. Lembrando que a avaliação do filtro é feita através do resíduo. E no quadro-05 geramos a matriz de estados do robô, que define sua trajetória, neste caso elíptica.

A figura 15 foi gerada pelo Matlab<sup>®</sup> através do código apresentado no quadro-04, representamos as posições: *desejada* (simbolicamente representada por - o) junto com um retângulo que representa nosso robô; *observada* (simbolicamente representada por - +) e *prevista pelo Filtro de Kalman* (simbolicamente representada por - \*), ambas com elipses à volta representando sua covariância.

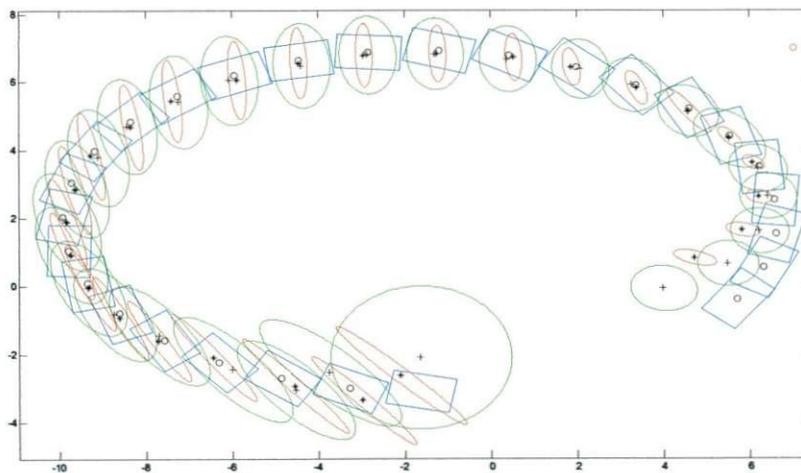


Figura 15 – Trajetória Elíptica: Desejada (o), Observada(+) e Prevista pelo Filtro de Kalman (\*), cortada antes de divergir.

É muito importante notar que para ambas as trajetórias [Figuras 14 e 15] a estimativa do percurso do filtro de Kalman e as medidas geradas pelo Matlab<sup>®</sup> acompanham muito bem as trajetórias propostas até 75% do percurso. A figura 15 foi criada com 23 iterações, parando antes de divergir; enquanto que a figura 16 foi criada com 30 iterações, e mostra claramente o ponto onde começam a divergir.

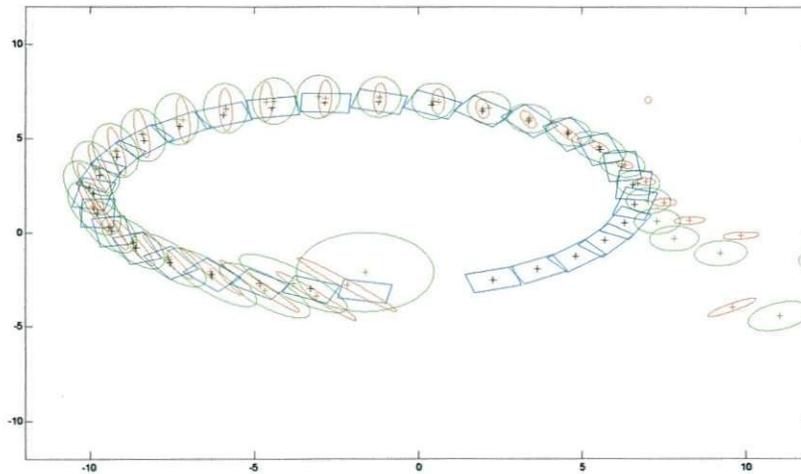


Figura 16 - Trajetória Elíptica: Desejada (azul), Observada (verde) e Prevista pelo Filtro de Kalman (vermelho), divergindo.

Como dissemos na seção 2 os ruídos são extremamente importantes para a descrição e controle do comportamento dos processos de estimação. Uma escolha não adequada dos ruídos de medição e previsão (no caso de simulações, já que na prática os ruídos de medição não são controláveis, mas podem bem identificados e compensados).

A explicação mais provável para a divergência no nosso caso é que as covariâncias calculadas dos erros nas estimativas podem estar muito pequenas. Por isso, o resultado da previsão através do modelo torna-se mais importante na estimativa do que as medidas subseqüentes, causando uma degradação significativa na precisão. Este efeito pode ser devido a uma infinidade de causas; em geral a imprecisão na modelagem ou mesmo o procedimento computacional adotado podem fazer com que o cálculo da matriz de covariância se torne um tanto quanto irreal.

## 4. Conclusão

O uso de robôs vem crescendo largamente dentro da indústria juntamente com a necessidade de eficiência de sistemas integrados de automação industrial. Com isso a mecatrônica ganha novas áreas de trabalho. Neste projeto nos focamos na robótica móvel, no controle da localização exata e desempenho correto das tarefas que estão destinadas aos Robôs.

Para isso, nos focamos no estudo e aplicação do Filtro de Kalman, ou seja, na implementação de algoritmos matemáticos estocásticos para o processamento de sinais discretos. Implementamos um sistema estimador para correção da posição do Robô baseado num Filtro de Kalman Estendido, para duas trajetórias, circular e elíptica. Também demos enfoque à questão da divergência, para o caso de sistemas cujas modelagens não atendem ao que era pedido, o que resulta em um projeto fraco.

Os conceitos relativos à estimação e correção de posição foram validados via simulação usando o Matlab<sup>®</sup>. Foi utilizado um conjunto de códigos para representar o sistema integrado interno de medição do manipulador.

Caracterizamos as medida matematicamente e demonstramos a importância da fusão de informação para reunir toda a informação válida. Como o modelo é não linear, foi utilizado um Filtro de Kalman Estendido para a fusão de informação de autocorrelação dos erros da medida e da estimação. Não há portanto, à priori, garantias teóricas de otimização e nem da convergência deste método.

Por isso, procuramos implementar um modelo de simulação, que permitisse condições próximas da realidade, para verificar o desempenho desta técnica. Um outro fator extremamente importante foi a caracterização das covariâncias dos sinais presentes no sistema.

Os resultados da simulação confirmaram que esta técnica é válida e promissora para o controle de trajetórias para robôs, já que corrigirem a sua localização considerando o erro mínimo quadrático das suas covariâncias.

A fusão de dados relativos à velocidade do robô móvel e do movimento do mesmo é conseguido, existindo convergência durante todo o processo por aproximadamente 75 à 80% dos percursos que propomos. O método apresentado não faz uma localização absoluta instantânea, mas medidas sucessivas levam a que o estado estimado convirja para o estado real do robô. Assim, com essa pesquisa e desenvolvimento, comprovamos à eficácia do Filtro de Kalman Estendido para o controle de trajetórias robóticas.



## 5. Referências Bibliográficas

- ANDREWS, A.P., GREWAL, M.S., **“Kalman Filtering: Theory and Practice Using MATLAB”** – Seconde Edition. EUA, John Wiley & Sons, Inc. -2001.
- DAUM, Fred, **“Kalman Filtering”** - The Electrical Engineering Handbook, p.p.435-445. EUA-2000.
- GROOVER, Mikel P., Weiss, Mitchell. **“Robótica: Tecnologia e Programação”**. McGrawHill, São Paulo. 1989.
- HAYKIN, Simon, **“Kalman Filtering and Neural Networks”**. EUA, John Wiley & Sons, Inc. -2001.
- HARVEY, A.C. **“Structural Time Series Models and the Kalman Filter”**. Cambridge University Press, Cambridge, 1989.
- HENRIQUES, R.V.B., **“Controle de posição de manipuladores: compensação através do filtro de Kalman”**. São Paulo, 1996. 115p. Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Eletrônica.
- JULIER, Simon J. & UHLMANN, Jeffery K., **“A New Extension of the Kalman Filter to nonlinear Systems.”** In *The Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls, Multi Sensor Fusion, Tracking and Resource Management II*, SPIE, 1997.
- REID, Ian., Term, Hilary. **“Estimation I”**. Class Notes. Dep. of Engineering Science. Oxford University, 2001.pp.01-36
- REID, Ian., Term, Hilary. **“Estimation II with a discrete time Kalman filter”**. Class Notes. Dep. of Engineering Science. Oxford University, 2001.pp.04-44.
- SIMON, D., **“Optimal state estimation: Kalman, H-infinity, and nonlinear approaches.”** John Wiley & Sons, 2006. (web site at: <http://academic.csuohio.edu/simond/estimation/>)
- SOUSA ,A.J.M. de; COSTA, P.J.G. da; MOREIRA, A.P.G.M., **“Sistema De Localização De Robôs Móveis Baseado em Filtro de Kalman Extendido”**. Paper - Faculdade de Engenharia, Universidade do Porto, Portugal. 2002.

### Websites:

<http://www.princeton.edu/~stengel/OptConEst.html>

[http://www.cs.unc.edu/~tracker/media/pdf/SIGGRAPH2001\\_CoursePack\\_08.pdf](http://www.cs.unc.edu/~tracker/media/pdf/SIGGRAPH2001_CoursePack_08.pdf)

[http://www.cs.unc.edu/~welch/media/pdf/kalman\\_intro.pdf](http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf)

[http://fr.wikipedia.org/wiki/Filtre\\_de\\_Kalman](http://fr.wikipedia.org/wiki/Filtre_de_Kalman)

<http://pt.wikipedia.org/wiki/Elipse>