

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/298352096>

# Filtr Kalmana – zastosowania w prostych układach sensorycznych.

Research · March 2016

DOI: 10.13140/RG.2.1.1336.4884

CITATIONS

8

READS

6,049

1 author:



Jan Kędzierski

Wrocław University of Science and Technology

16 PUBLICATIONS 150 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



EMYS - the first commercially available social robot who can teach foreign languages to kids aged 3-7. [View project](#)

# Filtr Kalmana - zastosowania w prostych układach sensorycznych.

Jan Kędzierski

Koło Naukowe Robotyków KoNaR.  
[www.konar.pwr.wroc.pl](http://www.konar.pwr.wroc.pl)

9 października 2007

## Spis treści

<b>1</b>	<b>Wstęp</b>	<b>2</b>
<b>2</b>	<b>Własności KF</b>	<b>2</b>
<b>3</b>	<b>Statystyka w KF</b>	<b>3</b>
3.1	Prawdopodobieństwo warunkowe. . . . .	3
3.2	Parametry rozkładu gęstości prawdopodobieństwa. . . . .	5
3.3	Przykład propagacji rozkładu prawdopodobieństwa. . . . .	6
<b>4</b>	<b>Filtr Kalmana</b>	<b>10</b>
4.1	Problem obserwatora . . . . .	11
4.2	Model procesu i pomiaru . . . . .	11
4.3	Dyskretny algorytm filtracji Kalmana . . . . .	12
<b>5</b>	<b>Rozszerzony Filtr Kalmana</b>	<b>13</b>
5.1	Przybliżenie procesu . . . . .	13
5.2	Estymacja procesu . . . . .	14
<b>6</b>	<b>Przykłady zastosowań FK.</b>	<b>16</b>
6.1	Liniowe systemy pomiarowe . . . . .	16
6.2	Nieliniowe systemy pomiarowe . . . . .	30

# 1 Wstęp

W 1960 r. R.E. Kalman opublikował słynną pracę, w której przedstawił metodę filtracji dynamicznej, która stała się jedną z najpopularniejszych metod estymacji statycznie optymalnej. Posługując się tym narzędziem, można wyznaczyć pomiarowo niedostępne zmienne jedynie na podstawie bieżących wartości wielkości pomiarowo dostępnych oraz znajomości modelu matematycznego łączącego ze sobą obydwie te grupy pomiarów.

Filtry Kalmana [1] (z ang. Kalman Filter w skrócie KF) znalazły zastosowanie w wielu dziedzinach nauki. Najczęściej są one stosowane w inżynierii, głównie w układach sensorycznych: robotów, samolotów oraz promów kosmicznych. Stosuje się je także w technikach komputerowych do przetwarzania obrazów oraz w ekonomii do prognozowania wskaźników gospodarczych. Metoda ta jest chętnie wykorzystywana w robotyce, gdzie układy percepcji otoczenia odgrywają istotną rolę. Są często jedynym źródłem informacji o środowisku, w którym znajduje się urządzenie.

W niniejszej pracy przedstawiono i wyjaśniono zasadę działania filtru, a także zawarto kilka praktycznych przykładów zastosowań w prostych układach pomiarowych.

# 2 Własności KF

W ramach wprowadzenia warto wspomnieć o własnościach filtru [2].

- KF jest optymalnym estymatorem, gdyż przy konkretnych założeniach może on spełniać pewne kryterium np. minimalizacja błędu średniokwadratowego<sup>1</sup> estymowanych parametrów.
- Kolejną cechą, która świadczy o optymalności filtru to fakt, że korzysta on z wszystkich dostępnych pomiarów bez względu na to, z jaką dokładnością i precyzją zostały one wykonane. Ostatecznie na ich podstawie dokonuje najlepszej estymacji stanu.
- KF jest algorytmem typu rekursywnego. Nie przechowuje on wszystkich danych z przeszłości i nie dokonuje on w każdym korku ich przeliczenia. Informacje są przetwarzane sukcesywnie, bazując na wartościach obliczonych w poprzednim kroku.
- Jest to algorytm przetwarzania danych. Znając wejście i wyjście systemu można uzyskać niedostępne (niemierzalne) wartości na podstawie dostępnych (mierzalnych) danych, np. z sensorów. W teorii systemów liniowych mówimy o obserwowalności układu.
- Metodę nazywamy filtrem, gdyż jest on optymalnym estymatorem stanu tzn., że uzyskamy możliwie optymalną wartość, na podstawie wielu pomiarów pochodzących z zaszumionego środowiska.

---

<sup>1</sup>błąd średniokwadratowy - mówi, o ile estymator różni się od wielkości, którą ma estymować.

### 3 Statystyka w KF

W robotyce filtrów Kalmana często używa się w systemach lokalizacji głównie autonomicznych robotów mobilnych [2]. Metoda ta dokonuje filtracji i fuzji pomiarów pochodzących z wielu sensorów. W procesie przetwarzania danych, uwzględniając prawdopodobieństwo wystąpienia błędu (szumy pomiaru, błędy systemu pomiarowego), szacuje się stan oraz prawdopodobieństwo jego wystąpienia. Zanim zostanie przedstawiona zasada działania filtru, niezbędne będzie przypomnienie kilku zagadnień ze statystyki [3].

#### 3.1 Prawdopodobieństwo warunkowe.

W tym rozdziale posłużono się przykładem jednowymiarowego problemu lokalizacji. Prawdopodobieństwo warunkowe będzie wartością prawdopodobieństwa, że robot znajduje się w stanie (na pozycji)  $x_k$  w czasie  $k$ , jeżeli wcześniej znajdował się w stanach  $z_0, z_1, \dots, z_k$  oraz podejmował akcję (przemieszczał się)  $a_0, a_1, \dots, a_k$ .

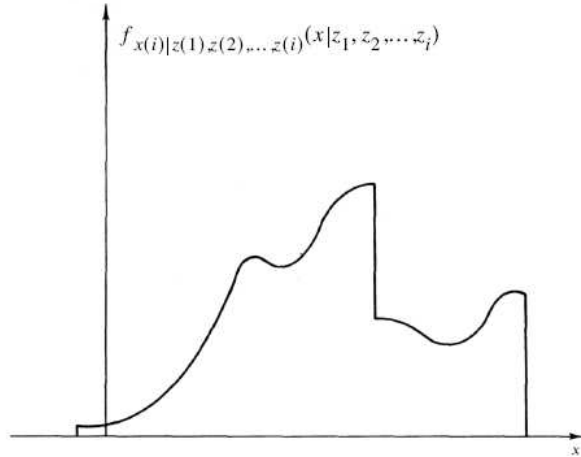
$$P(x_k | z_0, a_0, z_1, a_1, \dots, z_k, a_k) \quad (1)$$

W dalszych rozważaniach w zbiorze danych (stany, akcje) pominięto podzbiór akcji.

Rozkład prawdopodobieństwa warunkowego zapisuje się w postaci funkcji,

$$f_{x(k)|z(1),z(2),\dots,z(k)}(x|z_1, z_2, \dots, z_k) \quad (2)$$

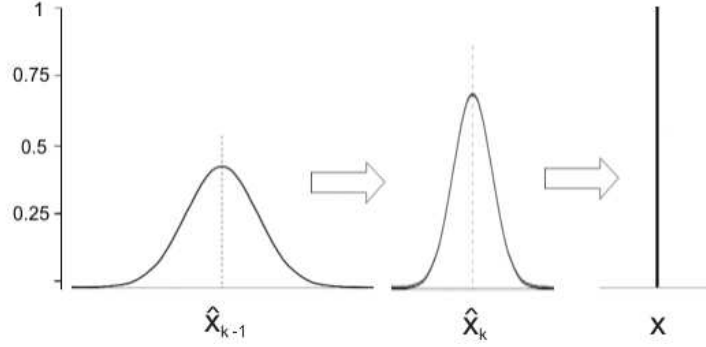
gdzie zmienna  $x \in X$ ,  $X$  jest zbiorem wszystkich możliwych stanów, dana  $z \in Z$ ,  $Z$  jest zbiorem danych (pomiarów). Innymi słowy na podstawie wyników pomiarów otrzymuje się w czasie  $k$  rozkład prawdopodobieństwa dla zmiennej  $x$ . Miejsce, w którym funkcja ma największą wartość jest stanem (pozycją) najbardziej prawdopodobnym [Rysunek 1]. Dane na podstawie, których wyznacza



Rysunek 1: Przykładowy rozkład prawdopodobieństwa warunkowego.

się prawdopodobieństwo warunkowe, mogą zawierać także mapy środowiska, które nie są w żaden sposób związane z pomiarami. Może to być np mapa

przeszkód, czyli obszary w których robot nigdy się nie znajdzie. W praktyce dąży się do uzyskania rozkładu zbliżonego do rozkładu tzw zmiennej pewnej. Ten z kolei ma postać delty Diraca [Rysunek 2].



Rysunek 2: Estymacja zmiennej  $\hat{x}$ .

Na potrzeby KF przedstawionych zostanie kilka konkretnych typów prawdopodobieństwa warunkowego, którymi posługiwano się w dalszej części pracy. Zapis  $Bel(x) = P(x|d_1, d_2, \dots, d_k)$  oznacza pewność (z ang. belief) w stanie  $x$ , czyli prawdopodobieństwo wystąpienia tego stanu na podstawie danych  $d_1, d_2, \dots, d_k$ .

- Prawdopodobieństwo wystąpienia stanu  $x_k$ , jeżeli we wcześniejszym kroku uzyskano stan  $x_{k-1}$ .

$$Bel(x_k) = P(x_k|x_{k-1}) \quad (3)$$

Naturalne jest to, że jeżeli robot porusza się z prędkością 1m/s, a znajdował się na pozycji 3m to najprawdopodobniej w następnej sekundzie powinien znajdować się w pozycji 4m. W praktyce znany jest tylko poprzedni stan, drugi zaś próbuje się oszacować. Szacowania dokonuje się na podstawie modelu ruchu, w praktyce kinematyki i dynamiki lub też wprost z nabytej wiedzy (roboty uczące się).

- Prawdopodobieństwo otrzymania pomiaru  $z_k$ , jeżeli obiekt znajduje się w stanie  $x_k$ .

$$Bel(z_k) = P(z_k|x_k) \quad (4)$$

W tym przypadku wykonuje się pomiar nieznannej wartości  $x$ . W praktyce najczęściej przyjmuje się błąd urządzenia pomiarowego lub jego poziom zakłóceń.

- Prawdopodobieństwo a priori.

$$Bel^-(x_k) = P(x_k|z_1, z_2, \dots, z_{k-1}) \quad (5)$$

Prawdopodobieństwo stanu  $x_k$  wyznaczone jedynie na podstawie wcześniejszych pomiarów. W ten sposób dokonujemy oszacowania stanu przed wykonaniem jego pomiaru.

- Prawdopodobieństwo a posteriori.

$$Bel^+(x_k) = P(x_k|z_1, z_2, \dots, z_k) \quad (6)$$

Prawdopodobieństwo stanu  $x_k$  na podstawie wszystkich pomiarów.

Każdy nowy stan szacuje się na podstawie poprzedniego stanu, a ten uzyskiwany jest dzięki dokonany w poprzednich krokach pomiarom:

$$P(x_k|x_{k-1}, z_1, z_2, \dots, z_{k-1}) \quad (7)$$

W filtrach Kalmana przyjmuje się założenie Markova, które mówi o tym, że żaden wcześniejszy pomiar nie wpływa na nowo uzyskany i vice versa. Zapiszmy więc ponownie równanie prawdopodobieństwa a priori (5) uwzględniając powyższe założenie:

$$Bel^-(x_k) = P(x_k|x_{k-1}, z_1, z_2, \dots, z_{k-1}) = P(x_k|x_{k-1}) \quad (8)$$

Następnie korzystając ze wzoru Bayesa <sup>2</sup> zapisujemy ponownie równanie prawdopodobieństwa (6) a posteriori:

$$\begin{aligned} Bel^+(x_k) &= \frac{P(x_k|x_{k-1}, z_1, \dots, z_{k-1})P(z_k|x_k, z_1, \dots, z_{k-1})}{P(z_k|z_1, \dots, z_{k-1})} \\ &= \frac{Bel^-(x_k)P(z_k|x_k, z_1, \dots, z_{k-1})}{P(z_k|z_1, \dots, z_{k-1})} \end{aligned} \quad (9)$$

i korzystając z założenia Markova, które mówi, że ostatni pomiar nie zależy od poprzednich i vice versa

$$P(z_k|x_k, z_1, z_2, \dots, z_{k-1}) = P(z_k|x_k) \quad (10)$$

zapisujemy:

$$Bel^+(x_k) = \frac{P(z_k|x_k)Bel^-(x_k)}{P(z_k|z_1, \dots, z_{k-1})} = P(x_k|z_1, \dots, z_k). \quad (11)$$

Można zauważyć, że powyższe działania przedstawiają rekursywny proces, niejako propagacji rozkładu prawdopodobieństwa warunkowego dla pożądanej wartości. Jest ona uwarunkowana danymi nadchodzącymi z pomiarów (czujników).

### 3.2 Parametry rozkładu gęstości prawdopodobieństwa.

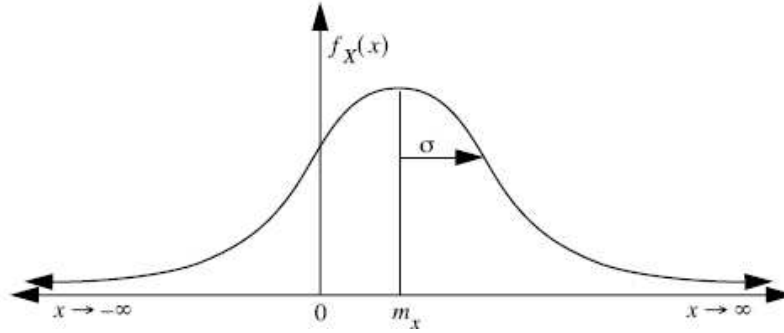
Na Rysunku 1 pokazano przykładowy rozkład gęstości prawdopodobieństwa. Każdy taki rozkład posiada następujące parametry:

- średnia (mean) - środek masy rozkładu
- dominanta (moda) - wartość  $x$ , w której funkcja osiąga maksimum - największe prawdopodobieństwo
- mediana (median) - wartość  $x$ , w szeregu uporządkowanym powyżej i poniżej, której znajduje się ta sama liczba, np. pomiarów.

---


$${}^2 P(B|A) = \frac{P(B)P(A|B)}{P(A)}$$

W filtrach Kalmana stosuje się, wspomnianą już wyżej, propagację gęstości prawdopodobieństwa warunkowego dla systemów opisanych modelem, w którym zarówno system jak i pomiar obarczone są szumami białymi<sup>3</sup> typu Gausa (Rysunek 3). W takich warunkach wszystkie parametry rozkładu (średnia, dominanta, mediana) pokrywają się, dając w każdym kroku szacowania wartość optymalną. Rozkład normalny definiuje się jako  $N(m_x, \sigma^2)$ , gdzie  $m_x$  - średnia,  $\sigma^2$  - wariancja (pierwiastek kwadratowy z wariancji to odchylenie standardowe  $\sigma$ ).



Rysunek 3: Rozkład prawdopodobieństwa typu Gausa (normalny).

### 3.3 Przykład propagacji rozkładu prawdopodobieństwa.

Peter S. Maybeck [3] w swojej pracy przedstawił prosty przykład, który w zasadzie będzie ilustrował zasadę działania KF.

Załóżmy, że dwóch żeglarzy wybrało się w rejs po morzu. Niestety zawiodły wszystkie urządzenia nawigacyjne. Jest środek nocy, a do dyspozycji mają tylko niebo pełne gwiazd. Pierwszy żeglarz wybiera zatem jedną z nich i wykonuje pomiar. Dla uproszczenia przyjęto, że jest on jednowymiarowy. Pomiar  $z_1$  wykonuje w czasie  $t_1$ . O jego precyzji mówi odchylenie standardowe  $\sigma_{z_1}$  (w statystyce często posługuje się wariancją  $\sigma_{z_1}^2$ ). A więc można określić prawdopodobieństwo ich pozycji  $x(t_1)$  w czasie  $t_1$  uwarunkowanego pomiarem  $z_1$  [Rysunek 4]. Widać, że pomiar nie jest zbyt dokładny, wykres jest łagodny, a odchylenie standardowe dosyć spore.

Estymowana wartość pozycji  $x$ , bazująca na wykresie z Rysunku 4 to:

$$\hat{x}(t_1) = z_1 \quad (12)$$

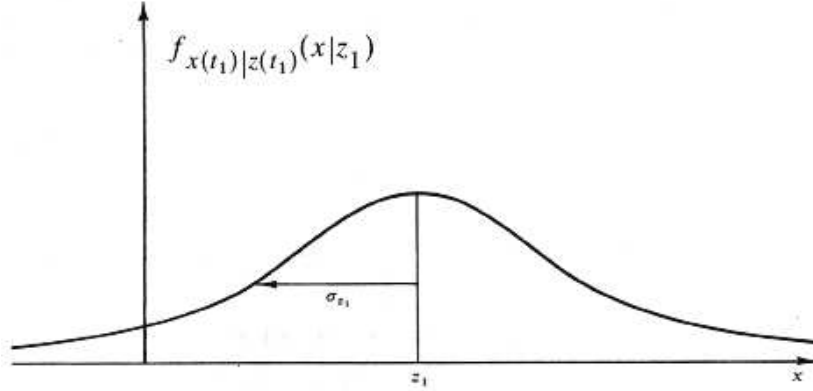
a jego wariancja, czyli błąd estymacji wynosi:

$$\sigma_x^2(t_1) = \sigma_{z_1}^2 \quad (13)$$

Kolejny pomiar wykonał drugi członek załogi. Jest on doświadczonym żeglarzem, a nawigacją zajmuje się już od dawna. Pomiar  $z_2$  zostanie wykonany w czasie  $t_2$  i obarczony będzie błędem, którego wariancja wynosi  $\sigma_{z_2}^2$ . Umiejętności drugiego sternika są o wiele większe, a zatem odchylenie standardowe jest znacznie mniejsze (Rysunek 5). Przyjęto także, że odcinek który przepłynęli w czasie  $t_2 - t_1$  nie

<sup>3</sup>Intensywność szumu białego teoretycznie jest statystycznie równomierna w całym paśmie od zera do nieskończoności, ale w praktyce przyjmuje się do rozważań tylko pewne zakresy częstotliwości.





Rysunek 4: Gęstość prawdopodobieństwa warunkowego w oparciu o  $z_1$ .

ma większego wpływu na pomiar. Ostatecznie uzyskali dwie wartości na podstawie których będzie można dokonać oszacowania pozycji (Rysunek 6). Średnią  $\mu$  nowego rozkładu oraz jego wariancję  $\sigma^2$  wyznacza się następująco:

$$\mu = \frac{\sigma_{z_2}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} z_1 + \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} z_2 \quad (14)$$

$$\frac{1}{\sigma^2} = \frac{1}{\sigma_{z_1}^2} + \frac{1}{\sigma_{z_2}^2} \quad (15)$$

Warto zauważyć, że wariancja  $\sigma^2$  jest mniejsza od  $\sigma_{z_1}^2$  i  $\sigma_{z_2}^2$ , czyli pewność pozycji zawsze wzrasta i jest większa od największej z pomiarów.

Uzyskane szacowanie w czasie  $t_2$  będzie teraz równe:

$$\hat{x}(t_2) = \mu \quad (16)$$

a jego wariancja:

$$\sigma_x^2(t_2) = \sigma_\mu^2 \quad (17)$$

Oczywiste wydaje się, że jeżeli wariancje  $\sigma_{z_1}^2$  i  $\sigma_{z_2}^2$  będą sobie równe, to wartość średnia  $\mu$  znajdzie się dokładnie w połowie odległości pomiędzy nimi.

Zapiszmy więc ponownie równanie (16) szacowanej pozycji w stanie  $t_2$ :

$$\hat{x}(t_2) = \frac{\sigma_{z_2}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} z_1 + \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} z_2 = z_1 + \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} (z_2 - z_1) \quad (18)$$

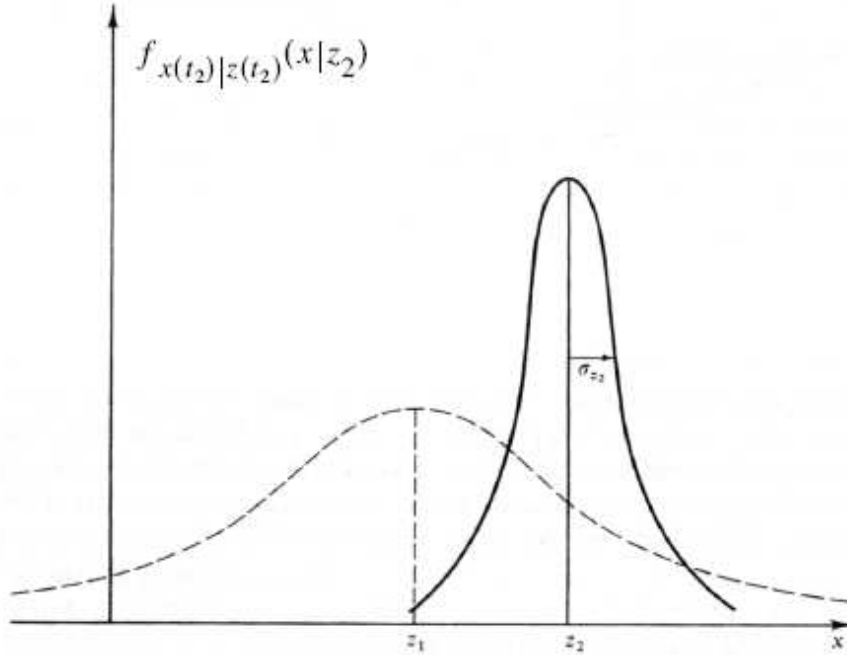
Zdefiniujmy współczynnik  $K$ , który nazywa się wzmocnieniem Kalmana:

$$K = \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} \quad (19)$$

Jeżeli przyjmie się, że  $z_1$  to poprzednio szacowany stan  $\hat{x}(t_1)$ , równanie (18) będzie miało postać:

$$\hat{x}(t_2) = \hat{x}(t_1) + K(z_2 - \hat{x}(t_1)) \quad (20)$$

Równanie to mówi o tym, jakie jest optymalne szacowanie stanu w czasie  $t_2$ . Jest ono równe oszacowanej wartości w czasie  $t_1$  plus korekcja uzyskana z różnicy



Rysunek 5: Gęstość prawdopodobieństwa warunkowego w oparciu o  $z_2$ .

pomiaru i poprzedniego szacowania, pomnożonej przez odpowiednią wartość wzmocnienia. Równanie wariancji w czasie  $t_2$  (17) będzie miało postać:

$$\sigma_x^2(t_2) = \sigma_x^2(t_1) - K(t_2)\sigma_x^2(t_1) = (1 - K(t_2))\sigma_x^2(t_1) \quad (21)$$

Wartości  $\hat{x}(t_2)$  oraz  $\sigma_x^2(t_2)$  ucieleśniają funkcję rozkładu prawdopodobieństwa warunkowego  $f_{x(t_2)|z(t_1), z(t_2)}(x|z_1, z_2)$  w czasie  $t_2$ .

Jak więc to działa w filtrze Kalmana? Kontynuując przykład morskiej żeglugi zapisane zostaną kolejno równania filtra dla wyżej opisanej sytuacji.

Po dokonaniu pierwszego pomiaru znane są już wartości początkowe  $\hat{x}(t_1)^+$  oraz  $\sigma_x^2(t_1)^+$ . W praktyce dobiera się je doświadczalnie lub przyjmuje wartości jednostkowe. W pierwszej fazie algorytmu, szacuje się wartości a priori pozycji  $x$  w czasie  $t_2$ . Wspomniano już, że przyjęto założenie, iż odcinek w którym przepłynął jacht jest niewielki i nie wpływa znacząco na pomiar. To znaczy, że w trakcie szacowania zakłada się, że jacht nadal znajduje się mniej więcej w tym samym miejscu:

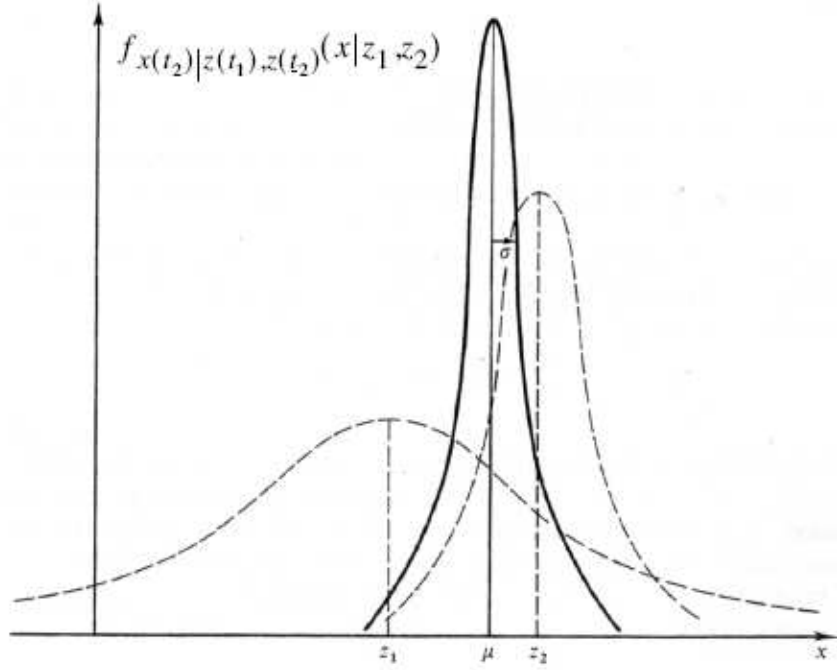
$$\hat{x}(t_2)^- = \hat{x}(t_1)^+ \quad (22)$$

$$\sigma_x^2(t_2)^- = \sigma_x^2(t_1)^+ \quad (23)$$

W drugiej fazie algorytmu wykonuje się kolejny pomiar oraz dokonuje korekcji oszacowanej w pierwszej fazie pozycji.

W tym celu oblicza się wzmocnienie  $K$ :

$$K(t_2) = \frac{\sigma_x^2(t_2)^-}{\sigma_x^2(t_2)^- + \sigma_{z_2}^2} \quad (24)$$



Rysunek 6: Gęstość prawdopodobieństwa warunkowego w oparciu o  $z_1$  i  $z_2$ .

gdzie wariancja  $\sigma_{z_2}^2$  to błąd pomiaru, który modeluje się jako biały szum Gaussowski o średniej  $z_2$  i odchyleniu standardowym  $\sigma_{z_2}$ . Następnie szacuje się nowe wartości pozycji i błędu.

$$\hat{x}(t_2)^+ = \hat{x}(t_2)^- + K(t_2)(z_2 - \hat{x}(t_2)^-) \quad (25)$$

$$\sigma_x^2(t_2)^+ = (1 - K(t_2))\sigma_x^2(t_2)^- \quad (26)$$

Z równań (15) i (23) warto zauważyć, że błąd  $\sigma_x^2$  będzie z czasem dążył do zera.

Przejdźmy teraz do opisu, w którym uwzględniono fakt, że jacht się przemieszcza. Załóżmy, że przy wykonywaniu predykcji, żeglarze oszacowali nowy stan, dodając do poprzedniego odcinek, który mogli przepłynąć. Niestety załoga nie posiada żadnego przyrządu do pomiaru prędkości z jaką płyną. Wyznaczyli ją jedynie na podstawie doświadczenia i obserwacji ruchu fal. Prędkość  $u$  będzie więc obciążona znaczącym błędem  $w$ , który modeluje się jako biały szum Gaussa o średniej zero i odchyleniu standardowym  $\sigma_w$ , świadczącym o niedokładności modelu ruchu (27) (z ang. motiom model).

$$\frac{dx}{dt} = u + w \quad (27)$$

Równania fazy predykcji będą wyglądały następująco:

$$\hat{x}(t_3)^- = \hat{x}(t_2)^+ + u(t_3 - t_2) \quad (28)$$

$$\sigma_x^2(t_3)^- = \sigma_x^2(t_2)^+ + \sigma_w^2(t_3 - t_2) \quad (29)$$

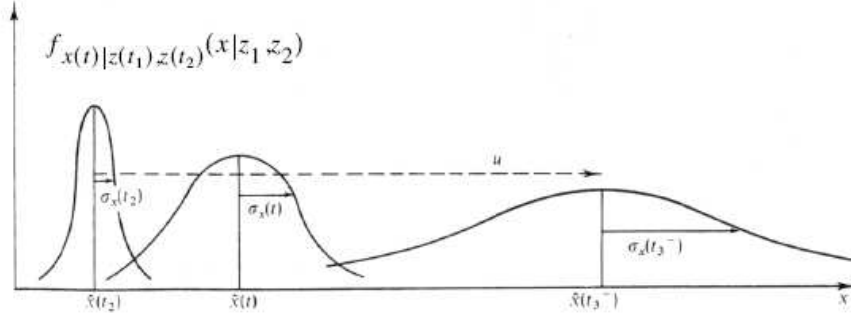
Druga faza (korekcja) pozostaje bez zmian.

$$K(t_3) = \frac{\sigma_x^2(t_3)^-}{\sigma_x^2(t_3)^- + \sigma_{z_3}^2} \quad (30)$$

$$\hat{x}(t_3)^+ = \hat{x}(t_3)^- + K(t_3)(z_3 - \hat{x}(t_3)^-) \quad (31)$$

$$\sigma_x^2(t_3)^+ = (1 - K(t_3))\sigma_x^2(t_3)^- \quad (32)$$

Tym razem błąd  $\sigma_x^2$  nie będzie dążył do zera, lecz zacznie zbiegać do pewnej wartości. Na rysunku (7) pokazano wpływ prędkości na rozkład prawdopodobieństwa warunkowego. Wraz z jej wzrostem, wzrasta także wariancja, gdyż traci się też pewność o aktualnej, zmierzonej w poprzednim kroku pozycji.



Rysunek 7: Wpływ prędkości na rozkład prawdopodobieństwa warunkowego.

Warto jeszcze zwrócić uwagę na kilka faktów wynikających z równań (30) i (31). Jeżeli wartość  $\sigma_z^2$  będzie duża (pomiaru są bardzo niedokładne) wzmocnienie  $K$  będzie małe. To oznacza, że w procesie filtracji głównie polega się na predykcji, a nie na pomiarach (szum pomiaru będzie mocno tłumiony).

$$\sigma_z^2 \rightarrow \infty, \quad K \rightarrow 0, \quad \hat{x}^+ \approx \hat{x}^- \quad (33)$$

Jeżeli  $\sigma_w^2$  będzie duża, to wartości  $\sigma_x^2$  i  $K$  też będą duże. Świadczy to o sporej niepewności tego, co się uzyskuje na wyjściu systemu. A zatem wynik będzie się głównie opierał na aktualnych pomiarach.

$$\sigma_w^2 \rightarrow \infty, \quad K \rightarrow 1, \quad \hat{x}^+ \approx z_x \quad (34)$$

## 4 Filtr Kalmana

Na co dzień posługujemy się czymś w rodzaju filtra Kalmana jednak nie zawsze jesteśmy tego świadomi. Estymujemy pewne wartości, takie jak godziny, ceny, czy też prędkości z jakimi się poruszamy. Przykładem może być poranny autobus, którym zwykle dojeżdżamy na uczelnię. Jeżeli od dłuższego czasu, z powodu np remontów na drogach autobus się spóźnia z czasem planowanego odjazdu. Innym przykładem może być zakup wymarzonego auta na wakacyjne wypady po atrakcyjnej cenie. Dziś każdy wie, że kabriolety osiągają szczytowe ceny zazwyczaj przed wakacjami. Dlaczego? Odpowiedź wydaje się oczywista. Takich przykładów można by przytoczyć więcej, ale istotne jest to, że tak naprawdę czynności, tych dokonujemy intuicyjnie. Jak więc zaimplementować podobny mechanizm w urządzeniach mikroprocesorowych?

Algorytm jest dosyć prosty i nie wymaga skomplikowanych, czasochłonnych obliczeń. Jedynie co należy wykonać to opisać proces jak i system pomiarowy odpowiednimi równaniami.

## 4.1 Problem obserwatora

Jednym z częstych problemów spotykanych w teorii systemów liniowych jest tzw. problem obserwatora [2]. Chodzi o to, aby potraktować system jak "czarną skrzynkę" i znając jedynie jej wyjścia, móc estymować jej wewnętrzne stany.

Może się zdarzyć tak, że na wyjściu obiektu otrzymujemy jedynie jakieś proporcje pomiędzy składowymi wektora stanu. Jeżeli więc nie będzie bezpośrednich relacji, pomiędzy pomiarem (wyjściem) a stanem wewnętrznym obiektu filtr będzie działał nieprawidłowo. Mówi się wtedy, że system jest nieobserwowalny.

## 4.2 Model procesu i pomiaru

Do opisu zarówno procesu jak i systemu pomiarowego stosują się modele matematyczne.

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad (35)$$

$$z_k = Hx_k + v_k \quad (36)$$

Pierwsze równanie różnicowe to model procesu, który częściowo jest deterministyczny a częściowo losowy. Jest to powiązanie poprzedniego stanu z aktualnym poprzez macierz  $A$ . Macierz  $B$  to wymuszenie stanu (sterowanie),  $w_k$  to tzw. szum procesu (część losowa). Drugie równanie to model pomiaru, gdzie  $H$  jest macierzą wiążącą pomiar ze stanem (wyjście filtru), a  $v_k$  to zakłócenia. Zarówno  $w_k$  jak i  $v_k$  reprezentują białe szумы Gaussa (Rysunek 3).

$$\begin{aligned} p(w) &\sim N(0, Q) \\ p(v) &\sim N(0, R) \end{aligned} \quad (37)$$

Podobnie zapis prawdopodobieństw warunkowych dla powyższych modeli będzie wyglądał następująco:

$$p(x_k | x_{k-1}) \sim N(Ax_{k-1} + Bu_k, Q) \quad (38)$$

$$p(z_k | x_k) \sim N(Hx_k, R) \quad (39)$$

Teraz zdefiniujmy błędy szacowania, w których  $\hat{x}^-$  to estymowany stan a priori uzyskany z procesu, a  $\hat{x}$  to estymowany stan a posteriori uwzględniający pomiar  $z_k$ .

$$e_k^- \equiv x_k - \hat{x}_k^- \quad (40)$$

$$e_k \equiv x_k - \hat{x}_k \quad (41)$$

gdzie  $e_k^-$  to błąd a priori, a  $e_k$  to błąd a posteriori. Są to różnice pomiędzy rzeczywistym stanem a wartością estymowaną. W praktyce rzeczywiste wartości stanu  $x_k$  nie są znane. Macierze kowariancji (zależności wariancji składowych wektora stanu) to:

$$P_k^- = E[e_k^-, e_k^-] \quad (42)$$

$$P_k = E[e_k, e_k] \quad (43)$$

gdzie  $P_k^-$  to macierz kowariancji a priori, a  $P_k$  to macierz kowariancji a posteriori. Ostatecznie można zapisać pewności estymowanych wartości stanu:

$$Bel(x_k)^- \sim N(\hat{x}^-, P_k^-) \quad (44)$$

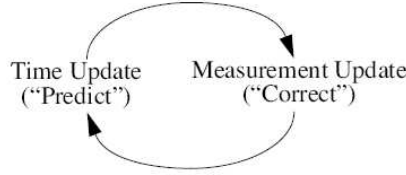
$$Bel(x_k) \sim N(\hat{x}, P_k) \quad (45)$$

oraz prawdopodobieństwo warunkowe dla stanu  $x_k$  bazujące na pomiarze  $z_k$ :

$$P(x_k|z_k) = N(\hat{x}, P_k) \quad (46)$$

### 4.3 Dyskretny algorytm filtracji Kalmana

Bazując na opisie z poprzedniego rozdziału powtórzmy ogólny zapis filtru dla naszego modelu (35). Filtr Kalmana jest dwufazowym rekursywnym algorytmem (Rysunek 8). Pierwsza faza algorytmu nazywana jest predykcją (z ang. predict). Równania wykonywane w trakcie tej fazy nazywane są aktualizacją czasową (z ang. time update). Druga faza nazywa się korekcją (z ang. correct), a jej równania to aktualizacja pomiarowa (z ang. measurement update). W trakcie predykcji, bazując na stanie z poprzedniego kroku, wyznacza się estymowaną wartość stanu  $\hat{x}$  oraz jego kowariancję i są to wartości a priori. Pomiar  $z$  w drugiej fazie jest pewną formą sprzężenia zwrotnego. Na jego podstawie dokonuje się wyznaczenia wartości a posteriori dla stanu i jego kowariancji.



Rysunek 8: Cykl dwufazowego algorytmu FK.

A oto równania pierwszej fazy:

$$\begin{aligned} \hat{x}_k^- &= A\hat{x}_{k-1} + Bu_{k-1} \\ P_k^- &= AP_{k-1}A^T + Q \end{aligned} \quad (47)$$

gdzie  $\hat{x}_k^-$  i  $P_k^-$  to prognozowane wartości stanu i kowariancji a priori,  $\hat{x}_{k-1}$  i  $P_{k-1}$  to optymalne szacowane wartości a posteriori wykonane w poprzednim kroku. Wzmocnienie Kalmana jest czymś w rodzaju wagi z jaką wpłynie faza korekcji na estymowany stan:

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \quad (48)$$

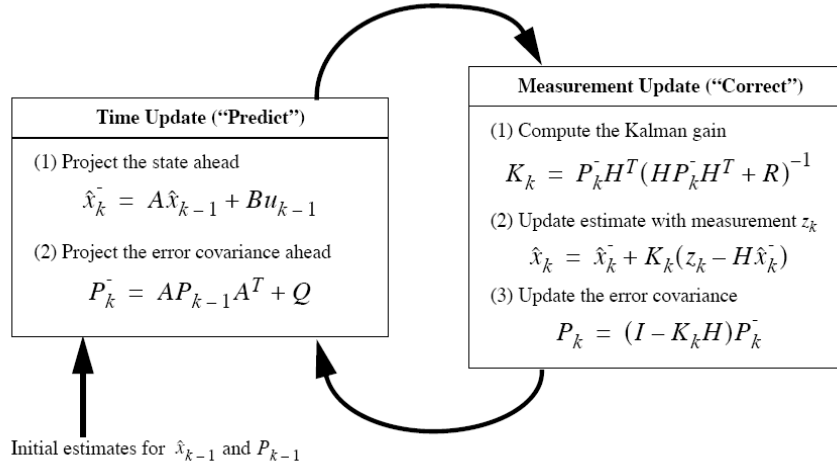
Równanie, które mówi jakie jest optymalne skorygowanie prognozy w czasie  $k$ , bazujące na wszystkich dotychczasowych pomiarach będzie miało postać:

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (49)$$

gdzie  $z_k$  to pomiar, a różnica  $(z_k - H\hat{x}_k^-)$  nazywa się *measurement innovation*. Pozostaje jeszcze skorygować macierz kowariancji:

$$P_k = (I - K_k H)P_k^- \quad (50)$$

gdzie  $I$  to macierz jednostkowa.



Rysunek 9: Kompletny algorytm FK.

## 5 Rozszerzony Filtr Kalmana

Tradycyjne filtry Kalmana stosuje się w dyskretnych procesach do optymalnej estymacji stanu. Zarówno proces, jak i związek pomiędzy pomiarem i procesem, opisuje się liniowymi równaniami (modelami). W przypadku nieliniowych równań stanu można skorzystać z tzw. rozszerzonego filtra Kalmana (z ang. Extended Kalman Filter w skrócie EKF), w którym równania stanu są zliniaryzowane poprzez zastąpienie nieliniowych funkcji ich rozwinięciami w szereg Taylora<sup>4</sup> w otoczeniu estymat.

### 5.1 Przybliżenie procesu

Dokonałmy zatem modyfikacji tradycyjnego filtra Kalmana. Równania stanu będą miały następującą postać:

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}) \quad (51)$$

$$z_k = h(x_k, v_k) \quad (52)$$

gdzie  $f$  to nieliniowa funkcja bazująca na poprzednim stanie, zawiera także sterowanie  $u_{k-1}$  oraz szum procesu  $w_{k-1}$ . Funkcja  $h_k$  to nieliniowa relacja pomiędzy stanem a pomiarem, zawierająca także błąd pomiaru  $v_k$ . Zarówno  $w_k$  jak i  $v_k$  reprezentują białe szumy Gaussa (Rysunek 3).

W praktyce nie znane są wartości szumów  $w_k$  i  $v_k$  w każdym kroku. Można zatem przybliżyć stan i pomiar nie uwzględniając szumów.

$$\tilde{x}_k = f(x_{k-1}, u_{k-1}, 0) \quad (53)$$

$$\tilde{z}_k = h(x_k, 0) \quad (54)$$

<sup>4</sup>szereg Taylora - przybliżenie w punkcie  $x_0$ , gdy liczymy tylko pierwszą pochodną to przybliżenie takie nazywamy liniowym,  $x \approx f(x_0) + \frac{f'(x)}{1}(x - x_0)$

Największą wadą EKF jest to, że rozkład zmiennej losowej po przejściu przez nieliniowe transformacje nie będzie już normalny (rozkład oszaleje :). Algoritm ten jest prostym estymatorem stanu ad hoc, który poprzez linearyzację, w sposób optymalny przybliża Bayesowskie reguły.

Uzyskane w procesie linearyzacji (szeregiem Taylora) równania stanu to:

$$x_k \approx \tilde{x}_k + A^J(x_{k-1} - \hat{x}_{k-1}) + W^J w_{k-1} \quad (55)$$

$$z_k \approx \tilde{z}_k + H^J(x_k - \tilde{x}_k) + V^J v_k \quad (56)$$

gdzie:

- $x_k$  i  $z_k$  — rzeczywiste, aktualne wektory stanu i pomiaru
- $\tilde{x}_k$  i  $\tilde{z}_k$  — przybliżone wartości stanu i pomiaru obliczone z nieliniowych równań 53 i 54
- $\hat{x}_k$  — oszacowany stan a posteriori (2 faza)
- $w_k$  i  $v_k$  — szum procesu i pomiaru ( $N(0, Q), N(0, R)$ )
- $A^J$  — macierz Jakobiego, pochodna funkcji  $f$  po  $x$

$$A_{[i,j]}^J = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\hat{x}_{k-1}, u_{k-1}, 0)$$

- $W^J$  — macierz Jakobiego, pochodna funkcji  $f$  po  $w$

$$W_{[i,j]}^J = \frac{\partial f_{[i]}}{\partial w_{[j]}}(\hat{x}_{k-1}, u_{k-1}, 0)$$

- $H^J$  — macierz Jakobiego, pochodna funkcji  $h$  po  $x$

$$H_{[i,j]}^J = \frac{\partial h_{[i]}}{\partial x_{[j]}}(\tilde{x}_k, 0)$$

- $V^J$  — macierz Jakobiego, pochodna funkcji  $h$  po  $v$

$$V_{[i,j]}^J = \frac{\partial h_{[i]}}{\partial v_{[j]}}(\tilde{x}_k, 0)$$

## 5.2 Estymacja procesu

Można spróbować wyznaczyć błędy przybliżeń i użyć, jeszcze raz równań Kalmana w celu ich oszacowania. A zatem zdefiniujemy rzeczywiste błędy przybliżeń:

$$\tilde{e}_{x_k} = x_k - \tilde{x}_k \quad (57)$$

$$\tilde{e}_{z_k} = z_k - \tilde{z}_k \quad (58)$$

W praktyce wartości  $x_k$  i  $z_k$  nie są znane. Podstawiając równania (55) i (56) do równań (57) i (58) otrzymano:

$$\tilde{e}_{x_k} = A^J(x_{k-1} - \hat{x}_{k-1}) + \epsilon_k \quad (59)$$

$$\tilde{e}_{z_k} = H^J(x_k - \tilde{x}_k) + \eta_k \quad (60)$$

, a w równaniu (60) dokonano podstawienia równania (57)

$$\tilde{e}_{x_k} = A^J(x_{k-1} - \hat{x}_{k-1}) + \epsilon_k \quad (61)$$

$$\tilde{e}_{z_k} = H^J \tilde{e}_{x_k} + \eta_k \quad (62)$$



gdzie  $\epsilon_k$  i  $\eta_k$  to nowe niezależne zmienne losowe z rozkładów  $p(\epsilon_k) \sim N(0, W^J Q_k (W^J)^T)$  i  $p(\eta_k) \sim N(0, V^J R_k (V^J)^T)$ . Traktując błąd  $\tilde{e}_{x_k}$  jako pewien stan, a  $\tilde{e}_{z_k}$  jako pewien pomiar można zauważyć, że równania (61) i (62) bardzo przypominają liniowe równania dla zwykłej wersji KF. Podpowiada to, że można wykorzystać aktualną wartość błędu pomiaru  $\tilde{e}_{z_k}$  (resztę szeregu Taylora przybliżającego  $z_k$ ) oraz hipotetycznie drugiego KF do estymowania rzeczywistego błędu prognozy (predykcji)  $\tilde{e}_{x_k}$ . Zdefiniujmy ten błąd jako  $\hat{e}_k$  czyli estymowany błąd w czasie  $k$ . Po przekształceniu równania (57), nieliniowe równanie procesu będzie miało teraz postać:

$$\hat{x}_k = \tilde{x}_k + \hat{e}_k \quad (63)$$

gdzie  $\hat{x}_k$ , to szacowany stan,  $\tilde{x}_k$  to przybliżony stan, a  $\hat{e}_k$  to szacowany błąd przybliżenia (reszta szeregu Taylora). Błąd ten szacujemy dzięki zastosowaniu hipotetycznie drugiego KF, wykorzystującego niejako pomiar błędu na podstawie równania (58).

Korzystając z równania Kalmana estymowany błąd przybliżenia na podstawie  $\tilde{e}_{z_k}$  można zapisać:

$$\hat{e}_k = \hat{e}_k^- + K_k(\tilde{e}_{z_k} - \hat{e}_k^-) \quad (64)$$

gdzie  $\hat{e}_k^-$ , to błąd jaki uzyskuje się w fazie predykcji. Gdy przyjmiemy, że  $\hat{e}_k^- = 0$ , bo taka powinna być poprawna prognoza, powyższe równanie będzie wyglądało następująco:

$$\hat{e}_k = K_k \tilde{e}_{z_k} \quad (65)$$

Błąd przybliżenia procesu będzie więc równy błędowi przybliżenia zależności pomiaru od stanu i pomnożonej przez wzmocnienie  $K_k$ . Podstawiając równanie (65) do równania (63) otrzymano:

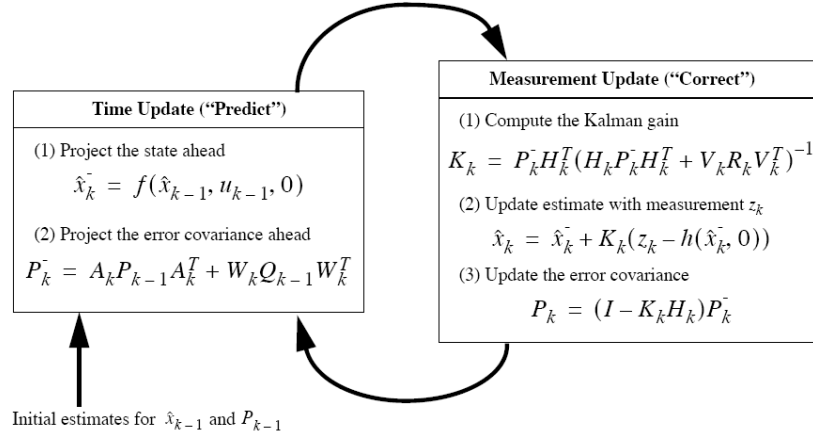
$$\hat{x}_k = \tilde{x}_k + K_k \tilde{e}_{z_k} \quad (66)$$

Ostatecznie korzystając z zależności (58) pokazano:

$$\hat{x}_k = \tilde{x}_k + K_k(z_k - \tilde{z}_k) \quad (67)$$

gdzie  $\tilde{x}_k$  i  $\tilde{z}_k$  wylicza się z równań (53) i (54).

Udowodniono, że w zasadzie nie jest potrzebny drugi FK, gdyż nie estymuje się błędów przybliżeń, a jedynie rzeczywisty stan. Wzmocnienie  $K_k$  oblicza się w standardowy sposób. Jakobiany  $A^J, W^J, H^J, V^J$  należy przeliczać w każdym cyklu działania algorytmu. Dlatego ich zapis powinien wyglądać  $A_t, W_t, H_t, V_t$ . Często w praktyce za  $W_t, V_t$  przyjmuje się 1, więc  $Q$  i  $R$  traktuje się jak w przypadku zwykłego KF, a w każdym kroku oblicza się jedynie  $A_t$  i  $H_t$ .



Rysunek 10: Kompletny algorytm EKF.

## 6 Przykłady zastosowań FK.

W niniejszym rozdziale zaprezentowano kilka przykładów z wykorzystaniem filtracji Kalmana. Wybrano je z myślą o zastosowaniu w układach zrobotyzowanych takich jak zintegrowane systemy nawigacyjne platform mobilnych, czy też pomiary odległości systemami złożonymi z wielu sensorów. Niektóre z nich zostały zilustrowane wykresami i fotografiami.

### 6.1 Liniowe systemy pomiarowe

**Przykład 1** *Pomiar zaszumionego napięcia.*

Przykład ten pomaga lepiej zrozumieć opisane powyżej działania. Jest to prosty pomiar stałego zaszumionego napięcia. Przyjęto, że szum jest typu białego. Ogólne równania systemu pomiarowego będą miały postać:

$$\begin{aligned} x_k &= A x_{k-1} + B u_{k-1} + w_{k-1} \\ z_k &= H x_k + v_k \end{aligned} \quad (68)$$

W systemie tym przyjmujemy, że  $A = 1$  ponieważ stan się nie zmienia, gdyż jest to pomiar stałego napięcia. Dlatego szacujemy, że w każdym kroku napięcie powinno mieć tę samą wartość. Nie ma żadnego wejścia sterującego zatem  $u = 0$ . Z uwagi na to, że zarówno pomiar jak i stan mają wymiar równy jeden macierz  $H = 1$ . Widać, że równania KF bardzo się uprościły:

Inicjacja:

$$\begin{aligned} x_0 &= 0 \\ P_0 &= 1 \\ \hat{x}_{k-1} &= x_0 \\ P_{k-1} &= P_0 \end{aligned} \quad (69)$$

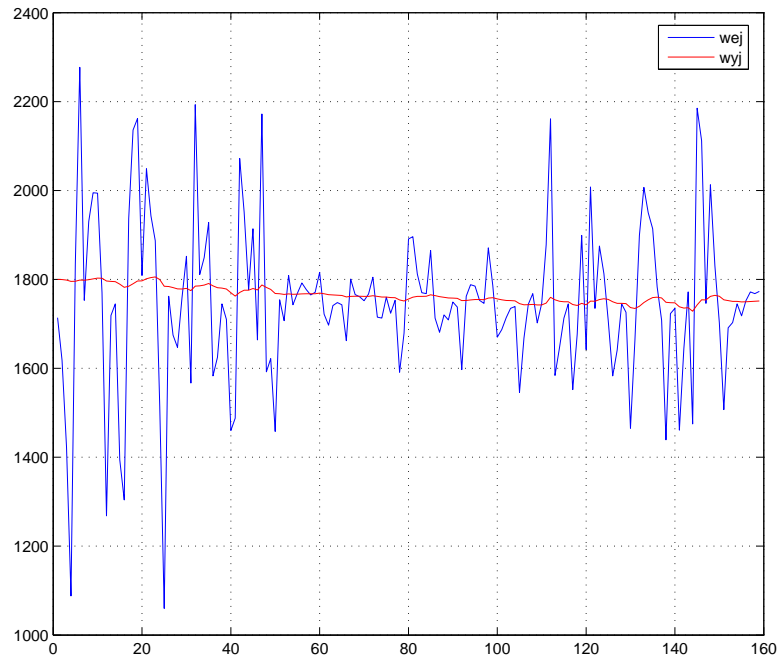
Predykcja:

$$\begin{aligned}\hat{x}_k^- &= \hat{x}_{k-1} \\ P_k^- &= P_{k-1} + Q\end{aligned}\tag{70}$$

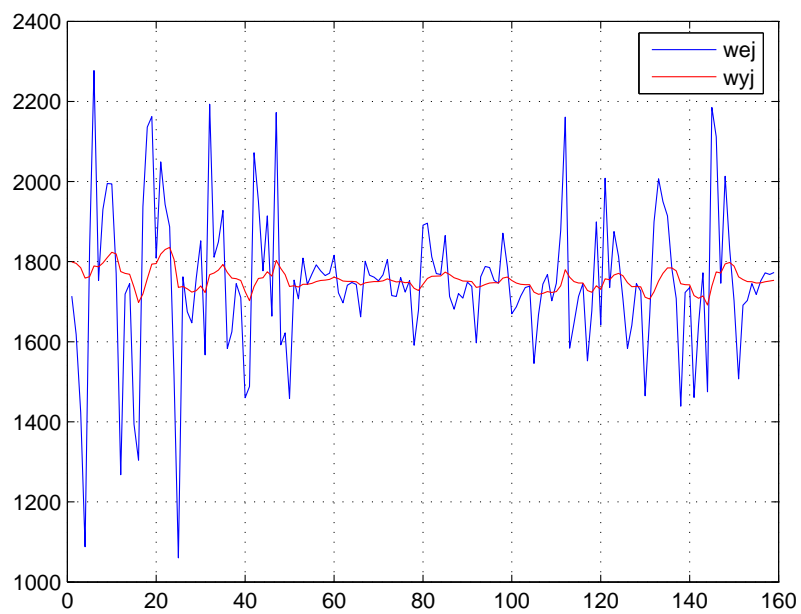
Korekcja:

$$\begin{aligned}K_k &= \frac{P_k^-}{P_k^- + R} = P_k^- (P_k^- + R)^{-1} \\ \hat{x}_k &= \hat{x}_k^- + K_k(z_k - \hat{x}_k^-) \\ P_k &= (1 - K_k)P_k^-\end{aligned}\tag{71}$$

Z pewnością wątpliwość budzi obecność wariancji  $Q$  w fazie predykcji. Wariancja  $Q$  powinna być równa zero. W podrozdziale 3.3 podano przykład, w którym uwzględniono znaczący błąd procesu, ponieważ jacht się przemieszczał w trakcie wykonywania serii pomiarów. Dlaczego pojawiła się niepewność procesu skoro jest on stały? Autorzy wielu prac o FK proponują przyjęcie za  $Q$ , dla stałych procesów, nawet bardzo małej wartości rzędu  $10e-5$  ponieważ lepiej się wtedy dostraja filtr.



Rysunek 11: Pomiar napięcia dla wariancji  $R = 0,01$  i  $Q = 10e-5$ .



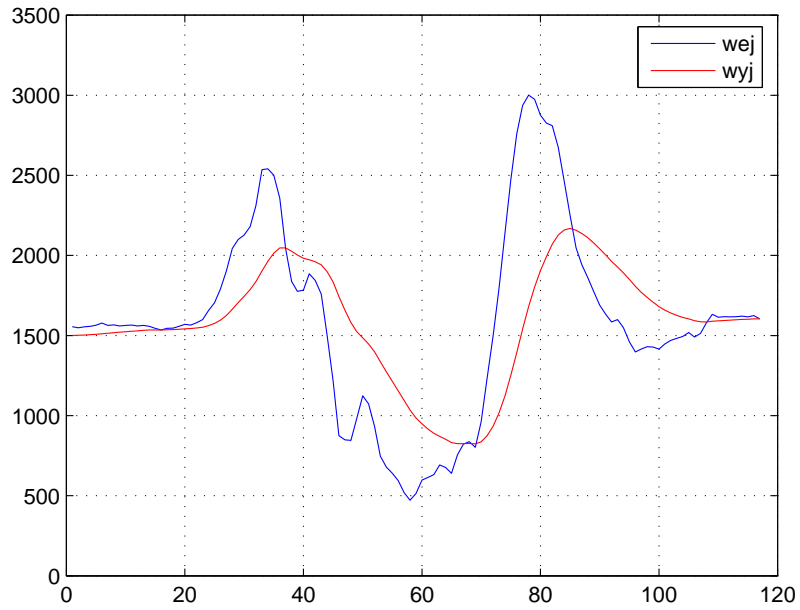
Rysunek 12: Pomiar napięcia dla wariancji  $R = 0,001$  i  $Q = 10e - 5$ .



Rysunek 13: Pomiar napięcia dla wariancji  $R = 0,0001$  i  $Q = 10e - 5$ .

### Przykład 2 Jednowymiarowy pomiar pozycji

Przykład ten ilustruje jak filtrować jednowymiarowy pomiar pozycji. Pomiary te mogą pochodzić, np z odbiornika GPS. Proces ten jest identyczny jak poprzedni. Równania mają taką samą postać :  $A = 1$ ,  $H = 1$ , tyle tylko, że w tym przypadku obiekt się porusza. Należy więc pamiętać, że za  $Q$  powinno się przyjąć dużą wartość.



Rysunek 14: Pomiar pozycji dla wariancji  $R = 0,1$  i  $Q = 0,001$ .

Na rysunku 14 przedstawiono wyniki filtracji. Widać, że pojawiło się spore przesunięcie fazowe sygnału. Jest to niepożądane zjawisko. Aby temu zaradzić, spróbujmy poddać estymacji nie tylko pozycję ale i prędkość. Wyniki działania przedstawiono na rysunku 15 Dla równań:

$$S_k = S_{k-1} + (V_{k-1} + w_{k-1})dt \quad (72)$$

$$V_k = V_k + v_k \quad (73)$$

gdzie  $S$ , to pozycja,  $V$ , to prędkość, macierz systemu ma postać:

$$\mathbf{A} = \begin{bmatrix} 1 & dt \\ 0 & 1 \end{bmatrix} \quad (74)$$

wektor stanu:

$$x = \begin{bmatrix} S \\ V \end{bmatrix} \quad (75)$$

Wyjście filtru:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad (76)$$

$R$  to nadal macierz  $1 \times 1$ , a macierz kowariancji  $Q$  ma wymiar  $2 \times 2$  i może być:

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} q \quad (77)$$

gdzie  $q$  to wariancja procesu.

$z_k$  - jednowymiarowy pomiar pozycji  $S$

Oto równania filtra dla powyższego systemu:

Inicjacja:

$$\begin{aligned} x_0 &= \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ P_0 &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} q \\ \hat{x}_{k-1} &= x_0 \\ P_{k-1} &= P_0 \end{aligned} \quad (78)$$

Predykcja:

$$\begin{aligned} \hat{x}_k^- &= A x_{k-1} \\ P_k^- &= A P_{k-1} A^T + Q \end{aligned} \quad (79)$$

Korekcja:

$$\begin{aligned} K_k &= \frac{P_k^- H^T}{H P_k^- H^T + R} = P_k^- H^T (H P_k^- H^T + R)^{-1} \\ \hat{x}_k &= \hat{x}_k^- + K_k (z_k - H \hat{x}_k^-) \\ P_k &= (I - K_k H) P_k^- \end{aligned} \quad (80)$$

Wymiar macierzy  $K$  jest  $1 \times 2$ .

Lepsze śledzenie sygnału można uzyskać, estymując także przyspieszenie. Wyniki działania filtra pokazano na rysunku 16. Macierz systemu wygląda wtedy następująco:

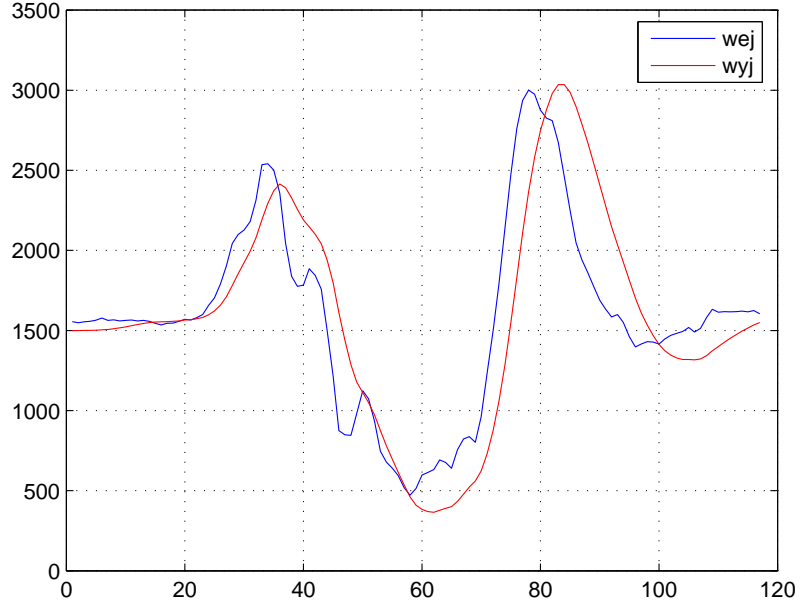
$$\mathbf{A} = \begin{bmatrix} 1 & dt & \frac{dt^2}{2} \\ 0 & 1 & dt \\ 0 & 0 & 1 \end{bmatrix} \quad (81)$$

wektor stanu:

$$\mathbf{x} = \begin{bmatrix} S \\ V \\ a \end{bmatrix} \quad (82)$$

gdzie  $S$  to droga,  $V$  to prędkość,  $a$  to przyspieszenie.

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \quad (83)$$



Rysunek 15: Estymacja pozycji i prędkości dla wariancji  $R = 100$  i  $Q = 0,01$ .

$R$  to nadal macierz  $1 \times 1$ , natomiast macierz kowariancji  $Q$  ma wymiar  $3 \times 3$  i może być:

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} q \quad (84)$$

gdzie  $q$  to wariancja procesu. Można także wyliczyć współczynniki (poszczególne wariancje):

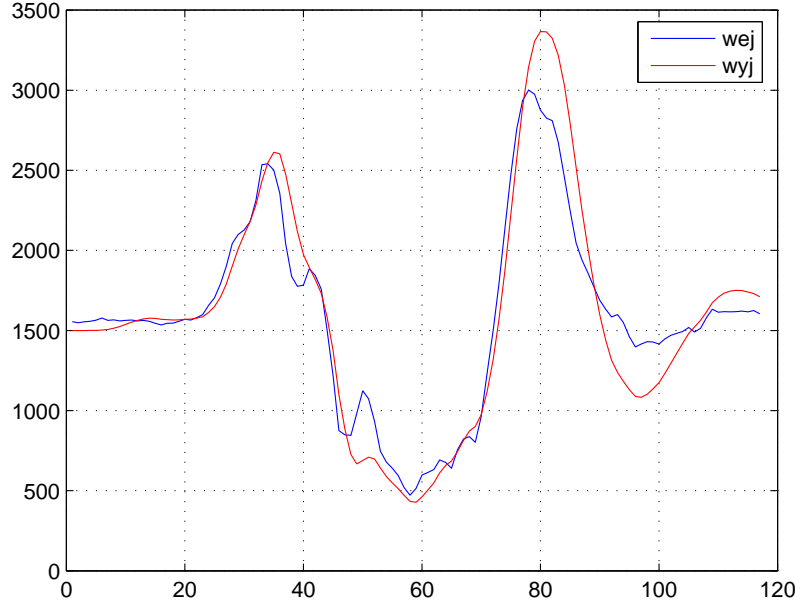
$$\mathbf{Q} = \begin{bmatrix} \frac{1}{20}dt^5 & \frac{1}{8}dt^4 & \frac{1}{6}dt^3 \\ \frac{1}{8}dt^4 & \frac{1}{3}dt^3 & \frac{1}{2}dt^2 \\ \frac{1}{6}dt^3 & \frac{1}{2}dt^2 & dt \end{bmatrix} q \quad (85)$$

$z_k$  - pomiar pozycji  $S$

Jeżeli system pomiarowy, zamiast GPS, zostanie wyposażony w czujnik przyspieszenia (akcelerometr), wtedy  $z_k$  będzie pomiarem przyspieszenia  $a$ , a więc macierz  $H$  należy zapisać w ten sposób:

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \quad (86)$$

reszta macierzy pozostaje bez zmian. Należy jednak pamiętać, że pojawi się niepożądane przesunięcie fazowe. Warto wtedy dodać wektora stanu tzw. szarp-



Rysunek 16: Estymacja pozycji, prędkości i przyspieszenia dla wariancji  $R = 2000$  i  $Q = 0,01$ .

nięcia (z ang. jerks). Macierz systemu będzie wtedy miała postać:

$$\mathbf{A} = \begin{bmatrix} 1 & dt & \frac{dt^2}{2} & \frac{dt^3}{6} \\ 0 & 1 & dt & \frac{dt^2}{2} \\ 0 & 0 & 1 & dt \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (87)$$

, a macierz wyjścia:

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \quad (88)$$

### Przykład 3 Wielowymiarowy pomiar pozycji

Tym razem system pomiarowy będzie dokonywał estymacji pozycji w trzech wymiarach. Wykorzystany zostanie ostatni wariant z poprzedniego przykładu, czyli estymacja pozycji, prędkości i przyspieszenia.

Macierz systemu będzie wyglądała:

$$\mathbf{A} = \begin{bmatrix} I & Idt & I\frac{dt^2}{2} \\ 0 & I & Idt \\ 0 & 0 & I \end{bmatrix} \quad (89)$$



gdzie  $I$  to macierz jednostkowa  $3 \times 3$ , wektor stanu:

$$\mathbf{x} = \begin{bmatrix} S_x \\ S_y \\ S_z \\ V_x \\ V_y \\ V_z \\ a_x \\ a_y \\ a_z \end{bmatrix} \quad (90)$$

gdzie  $S$ , to droga,  $V$  to prędkość,  $a$  to przyspieszenie wzdłuż poszczególnych osi  $xyz$ .

$$\mathbf{H} = \begin{bmatrix} I & 0 & 0 \end{bmatrix} \quad (91)$$

gdzie  $I$ , to macierz jednostkowa  $3 \times 3$ .  $Q$  to macierz kowariancji o wymiarze  $9 \times 9$  i może być:

$$\mathbf{Q} = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} q \quad (92)$$

gdzie  $q$  to wariancja procesu,  $I$  to macierz jednostkowa  $3 \times 3$ . Macierz  $R$  ma wymiar  $3 \times 3$  i wygląda następująco:

$$\mathbf{R} = \begin{bmatrix} r_x & 0 & 0 \\ 0 & r_y & 0 \\ 0 & 0 & r_z \end{bmatrix} \quad (93)$$

gdzie  $r$ , to wariancja poszczególnych pomiarów. Pomiar  $z_k$  trójwymiarowej pozycji  $S$  to:

$$\mathbf{z}_k = \begin{bmatrix} z_x \\ z_y \\ z_z \end{bmatrix} \quad (94)$$

Przykład implementacji filtra w  $C++$  przy użyciu biblioteki OpenCV<sup>5</sup>. Ilustruje on estymację dwuwymiarowej pozycji rozpoznanego na obrazie obiektu.

Należy na początku zadeklarować zmienne i struktury niezbędne do prawidłowej pracy algorytmu. Wektorem stanu jest pozycja  $x$  i  $y$  na dwuwymiarowym obrazie.

```
#include "cv.h"
#include "highgui.h"
// macierz systemu (zgodnie z systemem pomiarowym)
const float A[2][2] = {{ 1,0},{0,1}};
//macierz wyjścia (zgodnie z systemem pomiarowym)
const float H[2] = { 1,0};
```

<sup>5</sup>jest biblioteką, funkcji wykorzystywanych podczas obróbki obrazu, opartą o otwarty kod, a zapoczątkowaną przez Intel. <http://sourceforge.net/projects/opencvlibrary/>

```

//tworzenie struktury typu CvKalman
CvKalman* kalman = cvCreateKalman( 2, 1, 0 );
// wektor stanu
CvMat* state = cvCreateMat( 2, 1, CV_32FC1 );
// wektor pomiarów
CvMat* measurement = cvCreateMat( 2, 1, CV_32FC1 );

//wypełnienie struktury kalman
memcpy( kalman->transition_matrix->data.fl, A, sizeof(A));
memcpy( kalman->measurement_matrix->data.fl, H, sizeof(H));
// parametr Q
cvSetIdentity( kalman->process_noise_cov, cvRealScalar(1e-1) );
//parametr R
cvSetIdentity( kalman->measurement_noise_cov, cvRealScalar(1e-2) );
cvSetIdentity( kalman->error_cov_post, cvRealScalar(1));

```

Rozpoznany środek (np najjaśniejszy punkt) obiektu zapisano jako wektor dwuelementowy *pomiar*

```

// punkt - k-ta próbka obarczona błędem
CV_MAT_ELEM(*measurement,float,0,0)=(float)pomiar.x;
CV_MAT_ELEM(*measurement,float,1,0)=(float)pomiar.y;
// Predykcja
cvKalmanUpdateByTime(kalman);
// Korekcja
cvKalmanUpdateByMeasurement(kalman,measurement);
// zapis wyjścia filtru do wektora punkt
punkt.x=(int)kalman_red->PosterState[0];
punkt.y=(int)kalman_red->PosterState[1];

```

Więcej informacji można znaleźć na stronie  
<http://www710.univ-lyon1.fr/~bouakaz/OpenCV-0.9.5/docs/>

#### **Przykład 4** *Pomiar odchylenia kąтового.*

Jest to jeden z najczęściej spotykanych przykładów wykorzystania KF. W obiektach, które nie mają stałego punktu odniesienia pomiarów odchylenia dokonuje się przy pomocy inklinometrów, akcelerometrów lub żyroskopów. Niestety żaden z sensorów nie dostarcza dostatecznie dokładnej informacji o odchyleniu (obrocie) kątowym. W praktyce wykorzystuje się kilka czujników dla jednego systemu. Najciekawszym przypadkiem jest inklinometr lub akcelerometr wraz z żyroskopem.

Akcelerometry mierzą przyspieszenia statyczne, np grawitacja i zdolność tą wykorzystuje się do wyznaczenia odchylenia od pionu. Tanie, dostępne na rynku modele charakteryzują się sporymi szumami na wyjściu, dlatego zaleca się stosowanie jeszcze jednego czujnika. Żyroskopy, jak wiadomo, dostarczają informacji o prędkości obrotowej. Niestety ich wadą jest fakt, że z czasem ulegają dryftowi, pomiary stają się więc błędne. Filtr Kalmana doskonale niweluje niepożądane zjawisko dryftu oraz dokonuje przy tym fuzji pomiarów dostarczając dostatecznie dokładnej i nie zaszumionej informacji o odchyleniu (obrocie).

Równanie systemu ma postać:

$$\theta_k = \theta_{k-1} + (\omega_{k-1} - g_{bias})dt \quad (95)$$

gdzie  $\theta$  to odchylenie katowe,  $\omega$  to prędkość katowa, a  $g_{bias}$  to dryft żyroskopu. Tym razem jeden z pomiarów uwzględniony zostanie już w fazie predykcji. Prędkość katowa, odczytana z żyroskopu,  $\omega$  posłuży jako sterowanie  $u$ . Można więc zapisać jeszcze raz równanie systemu:

$$x_k = Ax_{k-1} + Bu \quad (96)$$

macierze:

$$\mathbf{A} = \begin{bmatrix} 1 & -dt \\ 0 & 1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} dt \\ 0 \end{bmatrix} \quad (97)$$

oraz wektor stanu:

$$x = \begin{bmatrix} \theta \\ g_{bias} \end{bmatrix} \quad (98)$$

Reszta macierzy wygląda jak w przypadku pomiaru jednym czujnikiem:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad (99)$$

$R$  to macierz  $1 \times 1$  (szum akcelerometru), macierz kowariancji  $Q$  ma wymiar  $2 \times 2$  i może być:

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} q \quad (100)$$

gdzie  $q$  to wariancja procesu (szum żyroskopu). Pomiar  $z_k$  to pomiar kąta  $\theta$  pochodzący z akcelerometru lub inklinometru.

Powyższy przykład pozwala estymować odchylenie katowe. Jeżeli potrzeba jest także informacja o prędkości katowej, należy nieco zmodyfikować macierze.

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & -dt \\ 0 & 0 & -1 \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} dt \\ 1 \\ 0 \end{bmatrix} \quad (101)$$

oraz wektor stanu:

$$x = \begin{bmatrix} \theta \\ \omega \\ g_{bias} \end{bmatrix} \quad (102)$$

Reszta macierzy wygląda jak w przypadku pomiaru jednym czujnikiem:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \quad (103)$$

$R$  to macierz  $1 \times 1$  (szum akcelerometru), macierz kowariancji  $Q$  ma wymiar  $3 \times 3$  i może być:

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} q \quad (104)$$

Poniżej przedstawiono przykład implementacji filtra dla dowolnego mikrokontrolera. W tym celu zadeklarowano dwie struktury *stan* oraz *kalm*. Pola tych struktur przechowują pomiary, wartości poszczególnych elementów macierzy  $P$ ,  $Q$  oraz wektora stanu  $x$ . Warto zwrócić uwagę na to, że nie ma konieczności obliczania wszystkich elementów macierzy  $P$ . W procesie filtracji biorą udział jedynie elementy  $P_{11}$ ,  $P_{13}$ ,  $P_{21}$ ,  $P_{31}$ ,  $P_{33}$ .

```
struct typ_stan {    /* Struktura stanu*/
float theta; /* odchylenie */
float omega; /* prędkość odchylenia */
float pomiar_theta; /* odchylenie */
float pomiar_omega; /* prędkość kątowa */
};
extern struct typ_stan stan;

struct typ_kalman { /* struktura filtra FK */
float Q; /* Wartości przekątnej macierzy Q */
float R; /* Wariancja pomiaru */
float theta; /* odchylenie */
float omega; /* prędkość kątowa */
float g_bias; /* dryft żyroskopu */
float P11; /* wartości macierzy kowariancji P */
float P13;
float P21;
float P31;
float P33;
float K1; /* wartości macierzy K */
float K2;
float K3;
};
extern struct typ_kalman kalm;
```

W programie głównym należy wszystkie pola zainicjować odpowiednimi wartościami.

```
Init()
{
// stan
stan.theta=0;
stan.omega=0;
stan.pomiar_theta=0;
stan.pomiar_omega=0;
// kalman
kalm.Q=0.0001;
kalm.R=1;
kalm.theta=0;
kalm.omega=0;
kalm.g_bias=0;
kalm.P11=R;
kalm.P13=0;
```

```

kalm.P21=0;
kalm.P31=0;
kalm.P33=0;
kalm.K1=0;
kalm.K2=0;
kalm.K3=0;

```

Teraz można przystąpić do rekursywnej pracy filtra. Do tego celu najlepiej wykorzystuje się wewnętrzne przerwania czasowe mikrokontrolera. Częstotliwość z jaką wykonuje się pomiary uzależniona jest od prędkości użytego przetwornika oraz wydajności jednostki centralnej. Wszystkie operacje arytmetyczne są wykonywane na zmiennych typu *float*, które są dość czasochłonne.

```

#define dt 0.02 // przerwanie co 20ms, f=50Hz

interrupt void int_TPUPITCHAN()
{
    //////////////////////////////////// pomiar
    stan.pomiar_theta=odczytaj_akcelerometry();
    stan.pomiar_omega=odczytaj_żyroskop();

    //////////////////////////////////// filtr kalmana
    // PREDYKCJA
    //
    //      |theta |      |1 0 -dt| |theta |      | dt|
    //      |omega |      |=0 0 -1 |*|omega |      |+| 1 |*gyro
    //      |q_bias |k      |=0 0  1 | |q_bias |k-1 | 0 |
    // xk apriori
    //
    kalm.theta=kalm.theta+(stan.pomiar_omega-kalm.g_bias)*dt;
    kalm.omega=stan.pomiar_omega-kalm.g_bias;
    // kalm.g_bias=kalm.g_bias;
    // -----
    // Pk=AP_{k-1}A^T+Q a priori
    kalm.P11=kalm.P11-kalm.P31*dt+kalm.P33*dt*dt-kalm.P13*dt+kalm.Q;
    kalm.P13=kalm.P13-kalm.P33*dt;
    kalm.P21=kalm.P33*dt-kalm.P31;
    kalm.P31=kalm.P31-kalm.P33*dt;
    kalm.P33=kalm.P33+kalm.Q;
    //
    ////////////////////////////////////
    // KOREKCJA
    // -----
    // Kk=PkH^T(HPkH^T+R)^-1
    kalm.K1=kalm.P11*(1/(kalm.P11+kalm.R));
    kalm.K2=kalm.P21*(1/(kalm.P11+kalm.R));
    kalm.K3=kalm.P31*(1/(kalm.P11+kalm.R));
    // -----
    // xk=xk-K(zk-Hxk) a posteriori
    kalm.theta=kalm.theta+kalm.K1*(stan.pomiar_theta-kalm.theta);
    kalm.omega=kalm.omega+kalm.K2*(stan.pomiar_theta-kalm.theta);

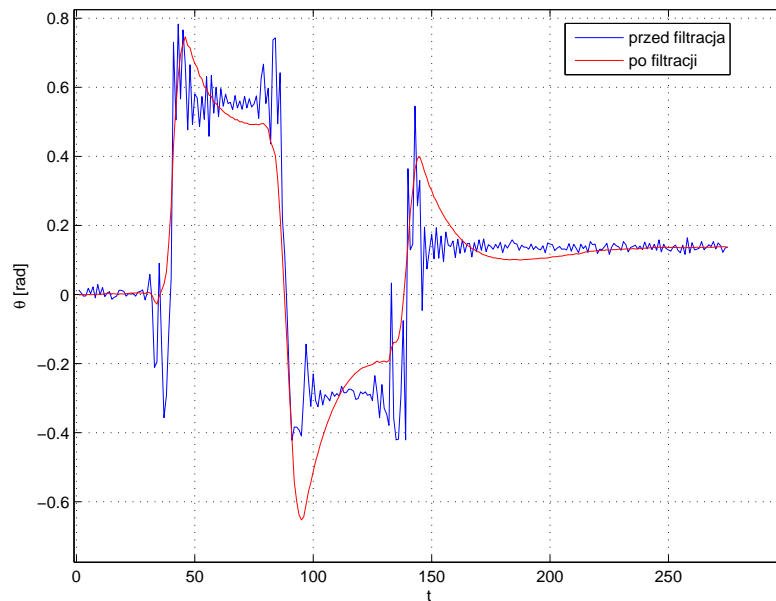
```

```

kalm.g_bias=kalm.g_bias+kalm.K3*(stan.pomiar_theta-kalm.theta);
// -----
// Pk=(1-KkH)Pk a posteriori
kalm.P11=(1-kalm.K1)*kalm.P11;
kalm.P13=(1-kalm.K1)*kalm.P13;
kalm.P21=kalm.P21-kalm.P11*kalm.K2;
kalm.P31=kalm.P31-kalm.P11*kalm.K3;
kalm.P33=kalm.P33-kalm.P13*kalm.K3;
//
////////////////////
// AKTUALIZACJA WEKTORA STANU
//
stan.theta=kalm.theta;
stan.omega=kalm.omega;
//
}

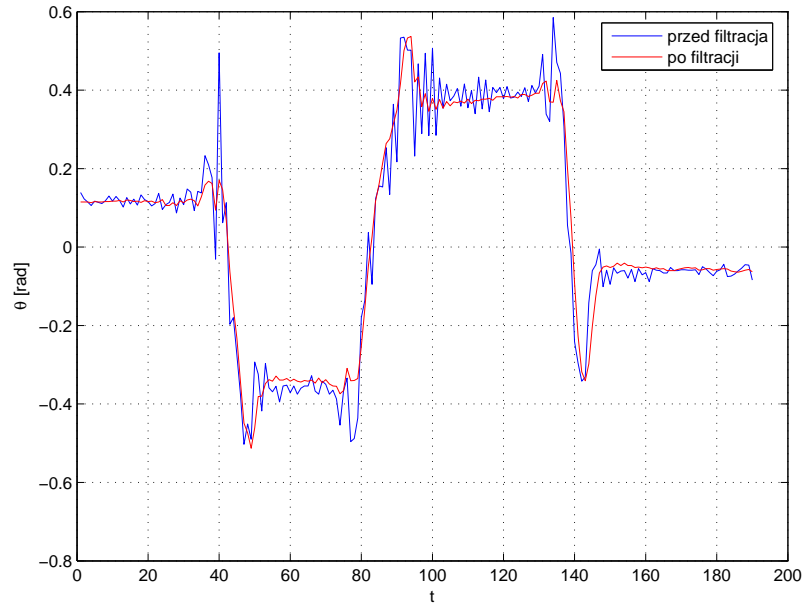
```

Układ pomiarowy przebadano dla różnych wartości  $Q$  i  $R$ . Wyniki filtracji pokazano na rysunkach 17, 18, 19.

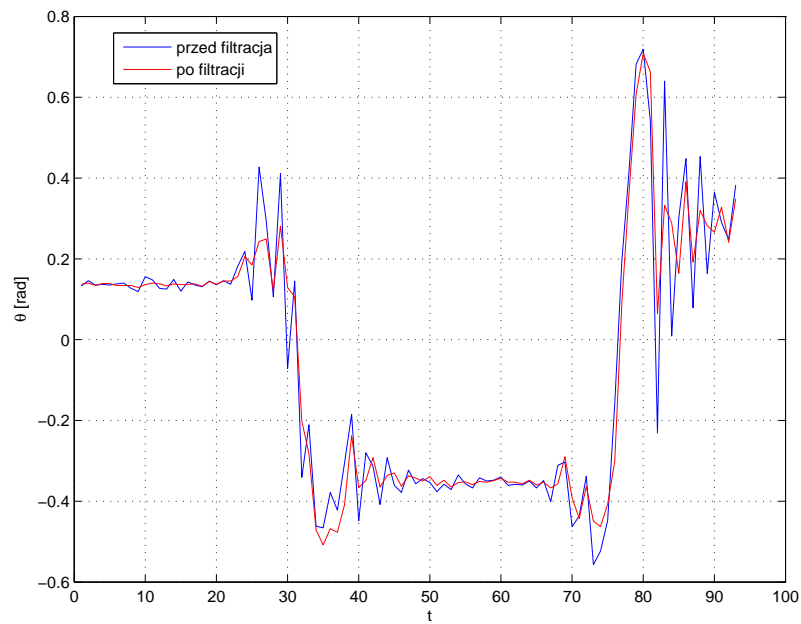


Rysunek 17: Pomiar odchylenia kąowego dla parametrów  $Q = 0.0001$  i  $R = 1$ .

Warto także rozważyć przypadek, w którym dokonuje się jedynie estymacji prędkości zmiany kąta. Można się wtedy posłużyć, w fazie korekcji, pomiarami z enkodera, który dostarczy informacji o prędkości kątowej. Powyższe równania



Rysunek 18: Pomiar odchylenia kąowego dla parametrów  $Q = 0.01$  i  $R = 10$ .



Rysunek 19: Pomiar odchylenia kąowego dla parametrów  $Q = 1$  i  $R = 1$ .

modyfikujemy analogicznie:

$$\begin{aligned}
\mathbf{A} &= \begin{bmatrix} 0 & -1 \\ 0 & 1 \end{bmatrix} \\
\mathbf{B} &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
x &= \begin{bmatrix} \omega \\ g_{bias} \end{bmatrix} \\
\mathbf{H} &= \begin{bmatrix} 1 & 0 \end{bmatrix} \\
\mathbf{Q} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} q
\end{aligned} \tag{105}$$

Wszystkie powyższe przypadki pomiaru odchylenia kąowego polegały na tym, że jeden czujnik wykorzystywano w fazie predykcji, drugi zaś w fazie korekcji. Czy można użyć dwóch sensorów w fazie korekcji? Można, o ile ma to w ogóle sens. Jak wspomniano wcześniej, jeden z czujników (żyroskop) z czasem ulega dryfowi. W fazie korekcji, dobrze jest się posługiwać czujnikami, które tylko szumią. Dryft nie należy do szumów (błędów losowych) jest to typ błędu systematycznego, który uwzględnia się w procesie pomiarowym, tak jak to miało miejsce w powyższych przykładach. Gdyby jednak posłużono się czujnikami np. enkoder oraz inklinometr równania mogłyby wyglądać następująco:

$$\mathbf{A} = \begin{bmatrix} 1 & dt \\ 0 & 1 \end{bmatrix} \tag{106}$$

sterowanie  $u$  jest równe 0 więc macierz  $B$  pomijamy.

$$\begin{aligned}
x &= \begin{bmatrix} \theta \\ \omega \end{bmatrix} \\
\mathbf{H} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\
\mathbf{Q} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} q
\end{aligned} \tag{107}$$

gdzie  $q$  to błąd procesu. Błędy czujników (odczytane np z ich dokumentacji) zapisuje się w macierzy  $R$ , która ma tym razem wymiar  $2 \times 2$ .

$$\mathbf{R} = \begin{bmatrix} R_{inklinometr} & 0 \\ 0 & R_{enkoder} \end{bmatrix} \tag{108}$$

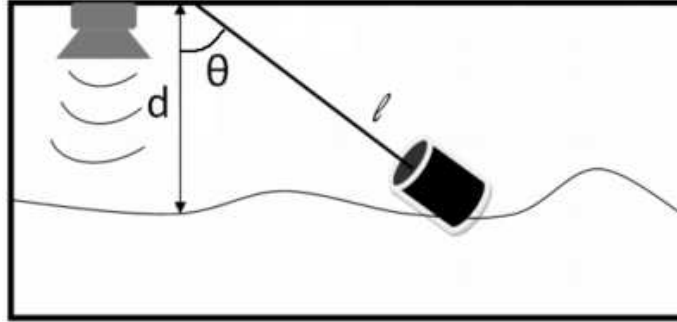
pomiar  $z_k$  nie będzie już jednowymiarowy tylko dwuwymiarowy:

$$\mathbf{z}_k = \begin{bmatrix} z_\theta \\ z_\omega \end{bmatrix} \tag{109}$$

## 6.2 Nieliniowe systemy pomiarowe

### Przykład 5 Pomiar poziomej cieczy





Rysunek 20: Pomiar poziomu cieczy w zbiorniku (sonar i pływak).

We wszystkich powyższych przykładach posługiwano się zwykłym filtrem Kalmana. Jak zatem w praktyce działa EKF? Jako przykład może posłużyć zbiornik z cieczą, na szczycie którego zainstalowano sonar do pomiaru poziomu oraz pływak wraz z enkoderem inkrementalnym. Z rysunku 20 widać, że system nie jest liniowy gdyż nieliniowa będzie zależność pomiędzy kątem zawieszonego pływaka a poziomem zmierzonym przez sonar.

W tym przykładzie, w fazie predykcji wykorzystuje się pomiar sonarem (jest on dosyć niedokładny). W trakcie korekcji posługuje się wyliczoną odległością na podstawie pomiaru kąta ramienia.

Nieliniowe równania systemu mają postać:

$$\begin{aligned} f(x, u) &= \begin{bmatrix} l \cos \theta \\ \theta \end{bmatrix} \\ h(x) &= \theta \end{aligned} \quad (110)$$

gdzie  $l$  to długość ramienia, na którym zawieszony jest pływak,  $\theta$  to kąt z jakim ramię się wychyliło. Wektor stanu to:

$$x = \begin{bmatrix} d \\ \theta \end{bmatrix} \quad (111)$$

gdzie  $d$  to poziom w zbiorniku. Jakobian  $A^J$  wyznacza się w każdym cyklu, gdyż zmienia się on w czasie.

$$\mathbf{A}^J = \begin{bmatrix} 0 & -l \sin \theta \\ 0 & 1 \end{bmatrix} \quad (112)$$

Reszta macierzy jest prosta i nie wymaga przeliczania w każdym cyklu:

$$\mathbf{H}^J = \begin{bmatrix} 0 & 1 \end{bmatrix} \quad (113)$$

$$\mathbf{W}^J = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (114)$$

$$\mathbf{V}^J = 1 \quad (115)$$

## Literatura

- [1] Greg Welch, Gary Bishop: *An Introduction to the Kalman Filter*. University of North Carolina at Chapel Hill Department of Computer Science Chapel Hill, NC 27599-3175
- [2] Rudy Negenborn: *Robot Localization and Kalman Filters. On finding your position in a noisy world*. Institute of Information and Computing Sciences in partial fulfilment of the requirements for the degree of Master of Science, specialized in Intelligent Systems
- [3] Peter S. Maybeck: *Stochastic models, estimation and control. Volume 1* Department of Electrical and Computer Engineering, Air Force Institute of Technology, Wright-Patterson Air Force Base Ohio.