

WPLYW IMPLEMENTACJI MODELU PRZEJŚCIA I MODELU POMIAROWEGO NA DZIAŁANIE FILTRU CZĄSTECZKOWEGO

Piotr Kozierski

Politechnika Poznańska, Wydział Elektryczny

Streszczenie. W artykule przedstawiono wpływ różnych realizacji modelu przejścia oraz modelu pomiarowego na działanie filtra cząsteczkowego. Zaproponowano przy tym kilka metod aproksymacji tych warunkowych funkcji gęstości prawdopodobieństwa.

Słowa kluczowe: filtr cząsteczkowy, obserwator stanu, model przejścia, model pomiarowy

EFFECT OF IMPLEMENTATION SYSTEM MODEL AND MEASUREMENT MODEL FOR PARTICLE FILTER WORKING

Abstract. The results of different implementations of system model and measurement model for particle filter are presented in the paper. Several approximation methods for those conditional probability density functions are proposed.

Keywords: particle filter, state observer, system model, measurement model

Wstęp

Model przejścia (ang. *System model*) oraz model pomiarowy (ang. *Measurement model*) w zadaniu filtra cząsteczkowego są przyjmowane za ogólnie znane [1, 3, 4]. Jednak różne sposoby ich określenia oraz implementacji mogą powodować rozbieżności w uzyskiwanych wynikach.

W pierwszym rozdziale została opisana zasada działania filtra cząsteczkowego. Drugi rozdział został poświęcony modelom przejścia i pomiarowemu oraz możliwości ich analitycznego przedstawienia. W trzecim rozdziale zaproponowano sposób aproksymacji obu modeli oraz przedstawiono możliwości polepszenia jakości działania filtra cząsteczkowego. Czwarty rozdział został poświęcony opisowi przeprowadzonych symulacji oraz uzyskanym wynikom. W piątym rozdziale zinterpretowano uzyskane wyniki.

1. Filtr cząsteczkowy

Zadaniem filtra cząsteczkowego jest (podobnie jak filtra Kalmana) estymacja stanu na podstawie obserwacji wyjścia obiektu (dla uproszczenia zapisu w rozważaniach zostanie pominięty wpływ sygnału wejściowego u_k).

Do rozwiązania tego problemu przyjmuje się, że znany jest model obiektu

$$\begin{aligned} x_k &= f_k(x_{k-1}, v_{k-1}) \\ y_k &= h_k(x_k, n_k) \end{aligned} \quad (1)$$

gdzie x_k to wartość zmiennej stanu w k -tej chwili czasowej, y_k to wartość obserwacji, v to szum wewnętrzny (ang. *process noise*), zaś n to szum pomiarowy (ang. *measurement noise*). Funkcje f_k oraz g_k są opisane nieliniowymi, zmiennymi w czasie równaniami [4].

Filtr cząsteczkowy jest alternatywną, nieparametryczną implementacją filtra Bayesa [5]

$$p(x_k | Y_k) = \frac{p(y_k | x_k) p(x_k | Y_{k-1})}{p(y_k | Y_{k-1})} \quad (2)$$

w której głównym pomysłem jest przedstawienie warunkowej funkcji gęstości prawdopodobieństwa (fgp) a posteriori $p(x_k | Y_k)$ (ang. *posterior*) za pomocą pewnej (skończonej) liczby próbek, z których każda ma określoną wagę [2, 3]. Pozostałe elementy równania (2) to wiarygodność $p(y_k | x_k)$ (ang. *likelihood*), fgp a priori $p(x_k | Y_{k-1})$ (ang. *prior*) oraz funkcja normująca $p(y_k | Y_{k-1})$ (ang. *evidence*).

W powyższym zapisie przyjmuje się, że

$$Y_k = \{y_1, y_2, \dots, y_k\} \quad (3)$$

Głównym celem algorytmu jest estymacja zmiennej stanu

$$\hat{x}_k = E[x_k | Y_k] = \int x_k p(x_k | Y_k) dx_k \quad (4)$$

co przy przedstawieniu fgp a posteriori za pomocą zbioru próbek można zapisać w postaci sumy

$$\hat{x}_k = E[x_k | Y_k] \approx \sum_{i=1}^N x_k^i q_k^i \quad (5)$$

Jak już zostało stwierdzone we wstępie, model przejścia $p(x_k | x_{k-1})$ oraz model pomiarowy $p(y_k | x_k)$ są ogólnie przyjmowane za znane, co jest związane ze znajomością funkcji i ich argumentów w (1).

Algorytm *Bootstrap filter* został zaproponowany w [3], a w [5] można znaleźć jego obszerny opis. Wielu autorów uważa go za pierwszy przykład filtra cząsteczkowego, choć tak naprawdę pomysł przedstawienia fgp w postaci zbioru cząstek powstał już w latach czterdziestych XX wieku [4].

Działanie algorytmu można zapisać następująco:

Algorytm 1 (Bootstrap filter)

- 1) Inicjalizacja algorytmu; $k=1$, wylosowanie N próbek z początkowego (znanego) rozkładu $p(x_1)$.
- 2) Przepuszczenie wszystkich próbek przez model przejścia $p(x_k | x_{k-1})$, czyli wylosowanie cząstek x_k^{*i} z fgp zależnego od wartości zmiennej stanu w chwili poprzedniej.
- 3) Obliczenie wagi każdej z cząstek zgodnie ze wzorem

$$q_k^i = \frac{p(y_k | x_k^{*i})}{\sum_{j=1}^N p(y_k | x_k^{*j})} \propto p(y_k | x_k^{*i}) \quad (6)$$

- 4) Normalizacja wag:

$$q_k^i = \frac{q_k^i}{\sum_{j=1}^N q_k^j} \quad (7)$$

- 5) Powtórne próbkowanie (ang. *resampling*); na podstawie zbioru cząstek x_k^{*i} i ich wag q_k^i losuje się N nowych cząstek, przy czym prawdopodobieństwo, że ta nowa cząsteczka będzie miała dokładnie wartość x_k^{*i} wynosi q_k^i :

$$\Pr\{x_k^j = x_k^{*i}\} = q_k^i \quad (8)$$

dla dowolnego j .

- 6) Rozkład cząstek informuje o obecnym wyglądzie fgp, z którego należy obliczyć wartość oczekiwaną zmiennej stanu wg wzoru (5); powrót do kroku 2.

Przedstawiony algorytm jest jednym z najprostszych, jednak posiada pewne wady, między innymi poprzez wykonanie kroku 5 dokładnie w taki sposób, jaki jest opisany (istnieje możliwość wylosowania 2 lub więcej identycznych próbek), jest pewne, że po pewnym czasie wszystkie cząsteczki będą miały dokładnie taką samą wartość. Aby tego uniknąć w implementacji przyjęto, że powtórne próbkowanie będzie przeprowadzane z odcinkowo ciągłej fgp o kształcie zbliżonym do tego otrzymanego na podstawie cząsteczek x_k^{*i} i ich wag q_k^i (przybliżenie wykonano za pomocą funkcji prostokątnej kształtem odpowiadającej histogramowi), nie jest to jednak przedmiot dalszych rozważań.

2. Modele przejścia i pomiarowy

Standardowym sposobem implementacji modelu przejścia jest zapisanie w kodzie jego funkcji. Załóżmy, że model (1) obiektu jest przedstawiony za pomocą równań stanu

$$\begin{aligned} x_k &= x_{k-1} + v_{k-1} \\ y_k &= x_k + n_k \end{aligned} \quad (9)$$

oraz, że szum wewnętrzny ma rozkład równomierny $v \sim U(0,2)$, natomiast szum pomiarowy ma rozkład normalny $n \sim N(0,1)$. Jeśli w poprzedniej chwili czasowej wartość zmiennej stanu wynosiła -1 , to w przypadku przepuszczania jej przez model przejścia $p(x_k|x_{k-1})$ będzie należało po prostu wylosować wartość nowej próbki z przedziału $(-1,1)$ o rozkładzie równomiernym.

Także w przypadku bardziej skomplikowanych modeli niż (9) można ograniczyć się do wylosowania próbki szumu ze znanego rozkładu, podstawić do wzoru i wykonać obliczenia.

W przypadku modelu pomiarowego wykorzystywanego w kroku 3 algorytmu jest już trochę trudniej zaimplementować sam wzór. W przypadku Modelu (9) nie jest to wprawdzie zbyt kłopotliwe, ponieważ można przekształcić drugie równanie do postaci

$$y_k - x_k = n_k \quad (10)$$

i na tej podstawie wykonać obliczenia na obserwacji modelu y_k oraz na wartości konkretnej cząsteczki x_k^{*i} . Po wykonaniu obliczeń należy poszukać wartości rozkładu funkcji szumu dla argumentu równego lewej stronie równania

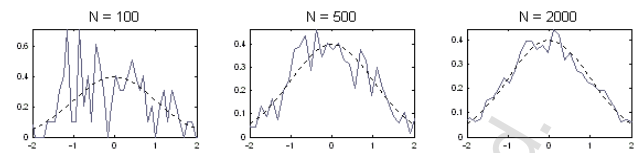
$$q_k^i = S(y_k - x_k^{*i}) \quad (11)$$

gdzie S to funkcja gęstości prawdopodobieństwa szumu.

Co jednak zrobić, jeśli nie da się analitycznie wyznaczyć równania na wartość wagi, bądź ma się do czynienia z obiektem rzeczywistym z możliwością odczytania wartości zmiennej stanu oraz wyjścia, ale np. w modelu brak informacji o konkretnych wartościach stałych (tzw. biała skrzynka – ang. *white box*), lub też chce się napisać uniwersalny algorytm, niezależny od modelu?

3. Aproksymacja obu modeli poprzez doświadczenie symulacyjne

W każdym z tych przypadków lepszym rozwiązaniem będzie aproksymacja obu modeli poprzez symulację – należy zebrać zestaw pomiarów zmiennej stanu oraz wyjścia obiektu i na tej podstawie oszacować warunkową fgp. Wraz ze wzrostem liczby pomiarów, na podstawie mocnego prawa wielkich liczb (mpwl), szacowana postać fgp będzie zbliżać się do jej prawdziwego kształtu, co zostało przedstawione na rys. 1.



Rys. 1. Wpływ wzrostu liczby próbek N na jakość odwzorowania fgp

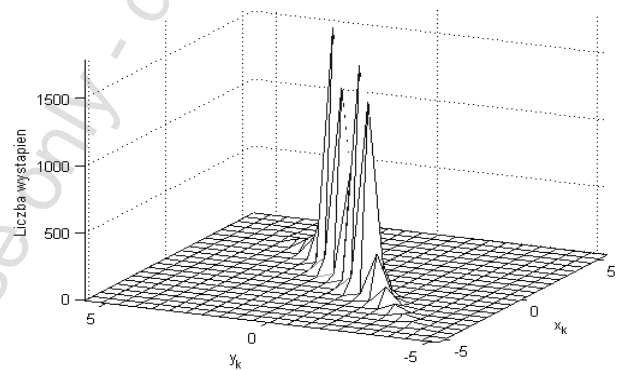
Jak można zaobserwować, dopiero przy aproksymacji danej fgp za pomocą 2 tysięcy i więcej próbek, rozkład przybliżony będzie przypominać ten rzeczywisty.

Oba modele, które będą aproksymowane zostaną tutaj rozpatrzone jako dwuwymiarowe funkcje, natomiast później należało brać pod uwagę tylko ich przekrój wzdłuż jednej z osi.

Okazuje się jednak, że takie podejście nie jest do końca dobre. Dla przykładu rozpatrzmy model obiektu

$$\begin{aligned} x_k &= 0.8x_{k-1} + \frac{\exp(0.1x_{k-1})}{0.1 + x_{k-1}^2} v_{k-1} \\ y_k &= x_k + n_k \end{aligned} \quad (12)$$

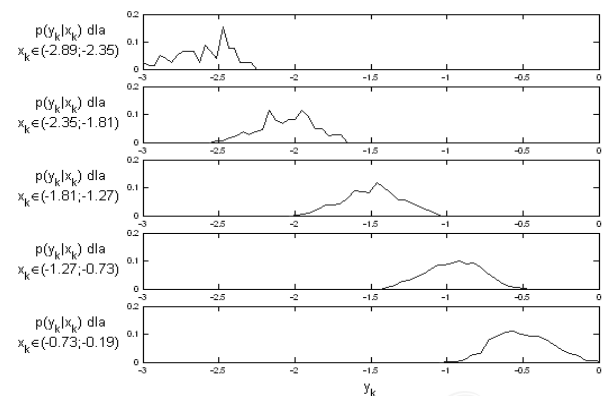
gdzie $v \sim N(0,0.2)$ oraz $n \sim N(0,0.1)$. Po przeprowadzeniu symulacji o długości 10^4 kroków otrzymano model pomiarowy przedstawiony na rys. 2.



Rys. 2. Graficzne przedstawienie modelu pomiarowego dla obiektu (12), dla 10000 kroków symulacji

Jak widać, taka reprezentacja modelu pomiarowego jest kompletnie nieprzydatna ze względu na bardzo dużą niedokładność – poza niewielką liczbą danych, większość wartości tej fgp jest równa zero.

Aby zwiększyć dokładność przechowywanych danych należy odrzucić zbędne informacje – zapisywać tylko wartości z zakresu, w którym prawdopodobieństwo jest większe od zera. Przy takim podejściu, zamiast jednej funkcji dwuwymiarowej zapisuje się zbiór funkcji cząstkowych jednowymiarowych, co zostało przedstawione na rys. 3.



Rys. 3. Fragment modelu pomiarowego zapisanego w postaci zbioru funkcji cząstkowych

Oczywistym jest fakt, że im dłuższy jest dostępny sygnał, tym modele są odwzorowane z większą dokładnością, a tym samym aby uzyskać lepszy efekt działania filtru, należy wydłużyć doświadczenie symulacyjne, na podstawie którego tworzone są model przejścia oraz model pomiarowy.

W celu dalszego polepszenia jakości działania filtru cząsteczkowego można zwiększyć liczbę funkcji opisujących model oraz liczbę próbek składających się na poszczególne funkcje. Przyjęto przy tym, że liczba funkcji cząstkowych L_f powinna być wprost proporcjonalna do pierwiastka drugiego stopnia (ponieważ modele są funkcjami dwuwymiarowymi) z liczby kroków symulacji, co można opisać wzorem

$$L_f = \frac{\sqrt{L_{sym}}}{W_p} \quad (13)$$

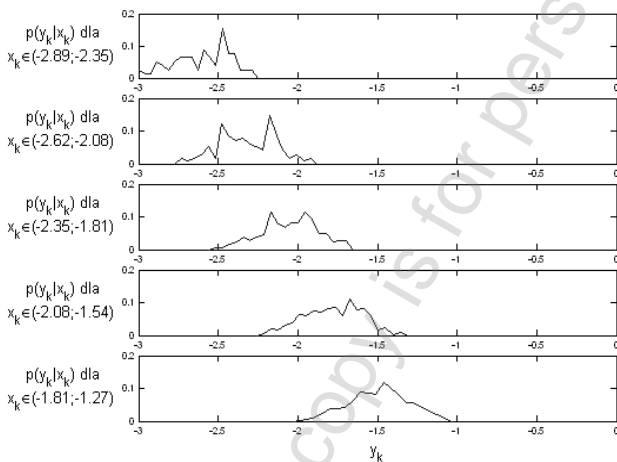
gdzie L_{sym} to długość sygnału, na podstawie którego są tworzone modele, zaś W_p to współczynnik, którego wpływ także został zbadany podczas symulacji.

Kolejnym aspektem mającym wpływ na jakość działania filtru cząsteczkowego jest sposób przybliżania wartości obu modeli podczas wykonywania algorytmu. Konkretnie chodzi o krok 2 (przepuszczenie cząsteczek przez model przejścia) oraz o krok 3 (przypisanie cząsteczce wagi na podstawie modelu pomiarowego) algorytmu. Sprawdzono dwie metody aproksymacji – „prostokątną” i „liniową”.

Na rys. 3 można zauważyć, że odległości pomiędzy kolejnymi funkcjami cząstkowymi (wzdłuż osi dla x_k) są stosunkowo duże w porównaniu do zakresu wartości y_k (zakres wartości y_k poszczególnych fgp jest w przybliżeniu równy 1, zaś odległość pomiędzy kolejnymi funkcjami cząstkowymi wynosi ponad 0.5). W związku z tym zaproponowano jeszcze jeden sposób polepszenia działania algorytmu polegający na „zagęszczeniu” modeli poprzez dodanie dodatkowych funkcji cząstkowych pomiędzy standardowymi.

Metoda ta różni się od zmiany wartości współczynnika W_p faktem, iż w tym przypadku szerokości poszczególnych przedziałów klasowych, pomimo większej liczby fgp, pozostały niezmiennione. Dzięki temu liczba pomiarów, na podstawie której tworzone są funkcje cząstkowe, jest taka sama, a nie mniejsza.

Wynik działania „zagęszczenia” przedstawiono na rys. 4



Rys. 4. Fragment „zagęszczonego” modelu pomiarowego zapisanego w postaci zbioru funkcji cząstkowych

Trzy pierwsze wykresy z rys. 3 odpowiadają wykresom 1, 3 i 5 z rys. 4. Jak można zaobserwować, przedziały wartości zmiennej stanu nachodzą na siebie, a tym samym jeden pomiar z doświadczenia symulacyjnego posłużył do obliczenia dwóch fgp, a nie jednej.

4. Symulacje

Do symulacji wykorzystano model obiektu (12). Aby porównanie poszczególnych rozwiązań z poprzedniego rozdziału było jak najbardziej miarodajne, wykorzystano identyczny sygnał testowy (wektory stanów i pomiarów) dla wszystkich przypadków.

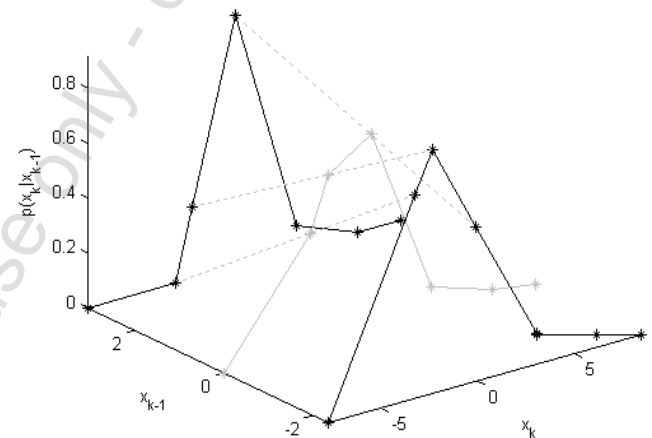
Następnie wykonano 4 doświadczenia symulacyjne o długości L_k równej 10^4 , 10^5 , 10^6 oraz 10^7 kroków – posłużyły one do stworzenia modelu przejścia oraz modelu pomiarowego.

Wartości parametru W_p , dla których zostały przeprowadzone obliczenia to:

$$W_p \in \{5, 10, 15, 20, 25, 30, 40, 50\} \quad (14)$$

Przyjęto, że liczba znaczących próbek każdej funkcji cząstkowej jest taka sama, jak liczba funkcji cząstkowych L_f .

Zaimplementowano także dwie różne metody aproksymacji w kroku 2 oraz kroku 3 algorytmu. W 2 kroku, w przypadku „aproksymacji prostokątnej” do losowania wybierano cząstkową fgp (model przejścia), która była najbliższa wartości cząsteczki x_{k-1} . W przypadku „aproksymacji liniowej” fgp wykorzystywana do wylosowania nowej cząsteczki była przybliżana w zależności od odległości między dwiema różnymi funkcjami cząstkowymi – im bliżej danej fgp znajdowała się cząstka x_{k-1} , tym bardziej wynik aproksymacji ją przypominał. Zostało to zobrazowane na rys. 5.



Rys. 5. Obliczanie fgp w 2 kroku algorytmu dla „aproksymacji liniowej”

Największą wadą tej metody jest fakt, że w każdej kolejnej chwili czasowej i dla każdej próbki trzeba tę fgp obliczać na nowo. Z kolei w przypadku „aproksymacji prostokątnej” – fgp są dostępne wprost z zapisanego w takiej postaci modelu, dystrybuanty (potrzebne do losowania z takiej fgp) także wystarczy wyznaczyć jeden raz na samym początku – w wynikach symulacji będzie widać znacznie większy czas obliczeń dla „aproksymacji liniowej”.

W 3 kroku algorytmu, wymagającym odczytania z modelu pomiarowego konkretnej wartości prawdopodobieństwa, wygląda to podobnie, jak w kroku poprzednim, z tym, że posiadając już odpowiednią fgp należy odczytać jej wartość dla konkretnego argumentu. Dla przybliżenia „prostokątnego” odczytywana jest wartość próbki znajdującej się najbliższej argumentu, zaś dla przybliżenia „liniowego” – poszukiwana jest dokładna wartość na prostej łączącej dwa sąsiednie punkty tej fgp.

Ostatnim parametrem, który ulegał zmianie podczas wszystkich symulacji było „zagęszczenie” opisane w poprzednim rozdziale – mogło być ono albo wyłączone ($ZAG=0$), albo włączone ($ZAG=1$). W pierwszym przypadku modele składały się z L_f funkcji cząstkowych, zaś w drugim przypadku liczba ta wzrastała do $2L_f-1$.

W tabeli 1 zostały przedstawione wyniki symulacji na które składają się czas symulacji t oraz średni kwadrat błędu wartości estymowanej dany wzorem

$$\delta = \frac{\sum_{k=1}^M (x_k - \hat{x}_k)^2}{M} \quad (15)$$

gdzie M to długość symulacji testowej (1000 iteracji), x_k to prawdziwa wartość zmiennej stanu w k -tej iteracji, zaś \hat{x}_k to estymowana wartość zmiennej stanu w k -tej iteracji. Liczbę cząsteczek przyjęto $N = 200$.

Tabela 1. Wyniki przeprowadzonych symulacji; δ – średni kwadrat błędu, t – czas symulacji [s]

L_{sym}	W_p	(L_f)	aproksymacja prostokątna		aproksymacja liniowa	
			ZAG=0	ZAG=1	ZAG=0	ZAG=1
10^4	50	(2)	$\delta=44.07$ $t=4.5$	$\delta=47.16$ $t=4.6$	$\delta=2.818$ $t=34.3$	$\delta=1.731$ $t=39.0$
	40	(3)	$\delta=2.320$ $t=9.0$	$\delta=2.328$ $t=5.5$	$\delta=0.489$ $t=42.2$	$\delta=0.158$ $t=47.2$
	30	(3)	$\delta=2.026$ $t=8.9$	$\delta=2.130$ $t=5.6$	$\delta=0.490$ $t=42.5$	$\delta=0.160$ $t=47.1$
	25	(4)	$\delta=0.368$ $t=6.3$	$\delta=0.635$ $t=6.1$	$\delta=0.301$ $t=44.1$	$\delta=0.124$ $t=49.7$
	20	(5)	$\delta=0.456$ $t=6.7$	$\delta=0.373$ $t=6.4$	$\delta=0.114$ $t=45.6$	$\delta=0.070$ $t=51.6$
	15	(7)	$\delta=0.311$ $t=7.1$	$\delta=0.269$ $t=7.1$	$\delta=0.099$ $t=49.4$	$\delta=0.043$ $t=57.0$
	10	(10)	$\delta=0.153$ $t=8.5$	$\delta=0.140$ $t=8.4$	$\delta=0.076$ $t=54.8$	$\delta=0.037$ $t=63.3$
	5	(20)	$\delta=0.147$ $t=12.9$	$\delta=0.130$ $t=12.8$	$\delta=0.108$ $t=75.3$	$\delta=0.096$ $t=89.2$
10^5	50	(6)	$\delta=0.616$ $t=6.9$	$\delta=0.530$ $t=6.7$	$\delta=0.188$ $t=54.3$	$\delta=0.080$ $t=53.8$
	40	(8)	$\delta=0.338$ $t=7.8$	$\delta=0.226$ $t=7.6$	$\delta=0.127$ $t=58.4$	$\delta=0.057$ $t=57.7$
	30	(11)	$\delta=0.173$ $t=9.2$	$\delta=0.132$ $t=8.9$	$\delta=0.075$ $t=64.5$	$\delta=0.026$ $t=63.9$
	25	(13)	$\delta=0.129$ $t=10.0$	$\delta=0.046$ $t=9.7$	$\delta=0.053$ $t=68.8$	$\delta=0.022$ $t=67.9$
	20	(16)	$\delta=0.090$ $t=11.0$	$\delta=0.058$ $t=10.8$	$\delta=0.060$ $t=78.0$	$\delta=0.016$ $t=76.8$
	15	(21)	$\delta=0.055$ $t=13.7$	$\delta=0.039$ $t=13.4$	$\delta=0.020$ $t=89.6$	$\delta=0.012$ $t=90.2$
	10	(32)	$\delta=0.061$ $t=18.7$	$\delta=0.033$ $t=18.7$	$\delta=0.012$ $t=114.1$	$\delta=0.009$ $t=113.6$
	5	(63)	$\delta=0.085$ $t=35.5$	$\delta=0.063$ $t=34.9$	$\delta=0.032$ $t=198.1$	$\delta=0.030$ $t=197.1$
10^6	50	(20)	$\delta=0.092$ $t=12.8$	$\delta=0.048$ $t=12.5$	$\delta=0.039$ $t=85.8$	$\delta=0.015$ $t=85.4$
	40	(25)	$\delta=0.023$ $t=15.0$	$\delta=0.015$ $t=14.7$	$\delta=0.022$ $t=98.0$	$\delta=0.013$ $t=97.5$
	30	(33)	$\delta=0.015$ $t=19.1$	$\delta=0.010$ $t=19.3$	$\delta=0.014$ $t=115.8$	$\delta=0.011$ $t=116.3$
	25	(40)	$\delta=0.012$ $t=22.7$	$\delta=0.010$ $t=22.3$	$\delta=0.011$ $t=131.8$	$\delta=0.010$ $t=131.7$
	20	(50)	$\delta=0.010$ $t=27.8$	$\delta=0.032$ $t=27.3$	$\delta=0.011$ $t=157.1$	$\delta=0.009$ $t=157.4$
	15	(67)	$\delta=0.032$ $t=35.4$	$\delta=0.031$ $t=35.7$	$\delta=0.032$ $t=207.7$	$\delta=0.009$ $t=203.7$
	10	(100)	$\delta=0.053$ $t=53.0$	$\delta=0.031$ $t=52.3$	$\delta=0.009$ $t=303.3$	$\delta=0.030$ $t=303.1$
	5	(200)	$\delta=0.032$ $t=116.1$	$\delta=0.053$ $t=118.3$	$\delta=0.031$ $t=663.9$	$\delta=0.030$ $t=660.6$
10^7	50	(63)	$\delta=0.009$ $t=33.9$	$\delta=0.033$ $t=34.3$	$\delta=0.010$ $t=194.9$	$\delta=0.031$ $t=198.0$
	40	(79)	$\delta=0.03$ $t=42.2$	$\delta=0.031$ $t=41.8$	$\delta=0.009$ $t=240.6$	$\delta=0.009$ $t=241.6$
	30	(105)	$\delta=0.009$ $t=54.5$	$\delta=0.009$ $t=54.7$	$\delta=0.010$ $t=316.8$	$\delta=0.009$ $t=316.2$
	25	(126)	$\delta=0.009$ $t=70.7$	$\delta=0.044$ $t=70.6$	$\delta=0.009$ $t=382.2$	$\delta=0.009$ $t=384.5$
	20	(158)	$\delta=0.009$ $t=88.9$	$\delta=0.009$ $t=89.0$	$\delta=0.009$ $t=491.2$	$\delta=0.009$ $t=493.0$
	15	(211)	$\delta=0.009$ $t=120.4$	$\delta=0.031$ $t=120.0$	$\delta=0.009$ $t=685.5$	$\delta=0.009$ $t=698.2$
	10	(316)	$\delta=0.055$ $t=192.2$	$\delta=0.031$ $t=188.1$	$\delta=0.032$ $t=1138.4$	$\delta=0.008$ $t=1126.5$
	5	(632)	$\delta=0.030$ $t=461.5$	$\delta=0.054$ $t=464.2$	$\delta=0.031$ $t=3140.5$	$\delta=0.030$ $t=2935.5$

5. Wnioski

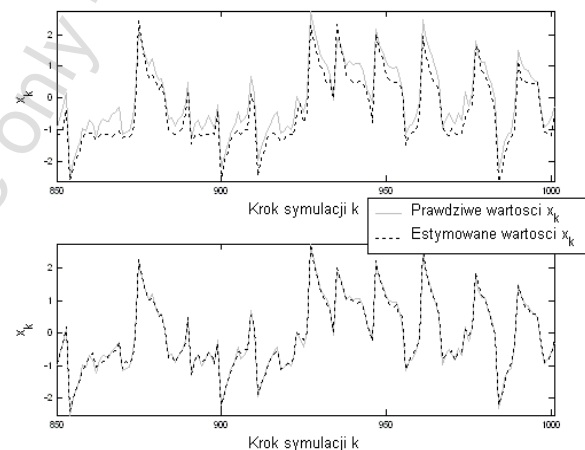
Analizując Tabelę 1 można zauważyć, że „aproksymacja liniowa” powoduje nieznaczne polepszenie działania algorytmu przy jednoczesnym kilkukrotnym wydłużeniu jego działania – „aproksymacja prostokątna” jest wystarczająco dobra do zastosowania jej w filtrze cząsteczkowym.

W przypadku „zagęszczenia” można stwierdzić, że praktycznie nie wpływa ono na czas symulacji, ale poprawa wyników jest bardzo niewielka. Jeśli pamięć, w której miałyby być przechowywane modele przejścia i pomiarowy (np. w mikrokontrolerze) jest dostatecznie duża można brać tę modyfikację pod uwagę, jednak poprawa jakości działania jest nieznaczna.

Znaczące różnice w działaniu algorytmu można zauważyć w związku ze zmianą parametrów L_{sym} oraz W_p . Można zaobserwować, że zbyt mała wartość W_p skutkuje pogorszeniem wyników symulacji – oznacza to, że zbyt duża liczba funkcji cząstkowych L_f jest niekorzystna, prawdopodobnie ze względu na fakt, że coraz więcej funkcji cząstkowych posiada wartość 0 nie tylko na skrajach, ale także w środku funkcji (zbyt mała liczba stanów w stosunku do liczby próbek składających się na oba modele).

Porównując czas działania symulacji dla podobnych wartości L_f stwierdzono, że czas ten zależy głównie od liczby funkcji cząstkowych i dla zbliżonych wartości L_f czas symulacji jest praktycznie identyczny. Jednocześnie porównując uzyskane wyniki dla symulacji o podobnym czasie widać, że lepsze rezultaty daje większe L_{sym} i większe W_p , co oczywiście jest związane z większą informacją niesioną przez dłuższe sygnały, z których zostały stworzone modele przejścia i pomiarowy.

Na rys. 6 dla porównania przedstawiono fragmenty symulacji – przebieg prawdziwej wartości zmiennej stanu oraz wartości estymowanej – aby zobrazować, jak się mają wartości z tabeli do odwzorowania kolejnych wartości x_k .



Rys. 6. Przykładowe przebiegi wartości zmiennej stanu oraz wartości estymowanej dla $\delta=0.275$ (pierwszy wykres) oraz $\delta=0.009$ (drugi wykres)

Literatura

- [1] Arulampalam S., Maskell S., Gordon N., Clapp T.: *A tutorial on Particle Filters for On-line Non-linear/Non-Gaussian Bayesian Tracking*. IEEE Proceedings on Signal Processing, Vol. 50, No. 2, 2002, s. 174-188.
- [2] Candy J.V.: *Bayesian signal processing*. WILEY, New Jersey 2009, s. 237-298.
- [3] Gordon N.J., Salmond N.J., Smith A.F.M.: *Novel approach to nonlinear/non-Gaussian Bayesian state estimation*. IEE Proceedings-F, Vol. 140, No. 2, 1993, s. 107-113.
- [4] Simon D.: *Optimal State Estimation*. WILEY-INTERSCIENCE, New Jersey 2006, s. 461-484.
- [5] Thrun S., Burgard W., Fox D.: *Probabilistic robotics*. MIT Press, Cambridge, MA, 2005, s. 67-90.

Piotr Kozierski

e-mail: piotr.kozierski@gmail.com

Student studiów stacjonarnych III st. na Politechnice Poznańskiej od 2010 roku. Obszar zainteresowań naukowych obejmuje identyfikację obiektów nieliniowych oraz filtry cząsteczkowe.

