

---

**СРАВНИТЕЛЕН АНАЛИЗ НА СЕДЕМ МЕТОДА  
ЗА ИНИЦИАЛИЗАЦИЯ НА ФИЛТЪР  
ЗА СЪПРОВОЖДАНЕ НА ТРАЕКТОРИИ  
НА ВЪЗДУХОПЛАВАТЕЛНИ СРЕДСТВА С BPNN**

**Мими Данева**

*mimidan@TU-Sofia.bg*

***Технически университет – София, катедра "Радиотехника"  
ул. Климент Охридски 8, София 1000, БЪЛГАРИЯ***

***Ключови думи:*** обработка на радиолокационна информация, невронни мрежи с обратно разпространение на грешката

***Резюме:*** В статията е представено експериментално сравнение между седем метода за инициализация на филтър за съпровождане на траектории на неманевриращи въздухоплавателни средства с многослойна невронна мрежа с обратно разпространение на грешката (Back-Propagation Neural Network, BPNN). Резултатите от изпълнение на BPNN филтъра при различните методи за инициализация са сравнени със стандартен рекурсивен филтър на Калман. За верификация на резултатите е направен Монте Карло експеримент. Представен е илюстративен пример на работата на BPNN филтъра при реални входни данни.

## **1. Увод**

Алгоритмите за филтрация с предсказване при обработка на траектории на въздухоплавателни средства (ВС), т.нар. филтри за съпровождане, заемат важно място в автоматизираните системи за обработка на радиолокационна информация (РЛИ) при управление на въздушното движение (УВД) [1], [2]. Те формират оценката на текущите и бъдещи кинематични параметри на ВС, които след това се използват в процеса на асоцииране на данните към наличните траектории на ВС, който представлява другата важна част на обработката на РЛИ. Като част от множеството алгоритми за съпровождане на траектории на множество цели (Multiple Target Tracking, МТТ), алгоритмите за филтрация с предсказване могат да бъдат разделени на две главни групи [1]: статистически и евристични. Филтрите за съпровождане, базирани на подхода на изкуствения интелект и в частност на невронните мрежи, принадлежат към втората група. Те се различават от

статистическите филтри по следните два основни признака [3]. Първо, те не се нуждаят от априорна информация за статистическите характеристики на входните данни и второ, за разлика от статистическите наблюдатели на състоянието те могат да формират оценката на вектора на състоянието на динамичната система (BC) без необходимост от използване на математически модел.

Кинематичният модел на неманевриращо BC е от втори ред (почти постоянна скорост на движение) и се дефинира посредством уравненията [1], [2]

$$(1) \quad \mathbf{X}(k+1) = \Phi \mathbf{X}(k) + \Gamma \mathbf{w}(k)$$

$$(2) \quad \mathbf{Z}(k) = \mathbf{H} \mathbf{X}(k) + \mathbf{v}(k)$$

където

$k$  – дискретно време;

$\mathbf{X}(k) = [\mathbf{X}_1^T(k) \ \mathbf{X}_2^T(k) \ \mathbf{X}_3^T(k)]$  – вектор на състоянието на динамичната система (BC) с компоненти векторите на състоянието  $\mathbf{X}_i = [\eta_i, \dot{\eta}_i]$  за координата  $i$ ,  $i=1,2,3$ . Всеки вектор  $\mathbf{X}_i$  съдържа позиция  $\eta_i$  и скорост  $\dot{\eta}_i$ . Символът  $i$  означава една от координатите на BC и се използва за простота на означенията.

Матриците  $\mathbf{w}$  и  $\mathbf{v}$  (всяка с размерност 3) представят взаимно некорелирани стохастични процеси, всеки с нулева средна стойност, известна дисперсия и некорелиран с  $\mathbf{X}(0)$ . С  $\mathbf{w}(k)$  се моделират случайните промени в скоростта на полета, а с  $\mathbf{v}(k)$  – грешките на РЛС при измерване на координатите. Ковариационните матрици на шумовете и системните матрици на модела са както следва [1]

$$\mathbf{Q} = [q_{ij}] = E\{\mathbf{w}(k) \mathbf{w}^T(k)\}; \quad \mathbf{R} = [r_{ij}] = E\{\mathbf{v}(k) \mathbf{v}^T(k)\};$$

$$\Phi = \text{diag}[\Phi_2 \ \Phi_2 \ \Phi_2]; \quad \Gamma = \text{diag}[\Gamma_2 \ \Gamma_2 \ \Gamma_2]; \quad \mathbf{H} = \text{diag}[\mathbf{H}_2 \ \mathbf{H}_2 \ \mathbf{H}_2]$$

където с горен индекс  $T$  е означен оператор за транспониране и

$$\Phi_2 = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}; \quad \Gamma_2 = \begin{bmatrix} T^2/2 \\ T \end{bmatrix}; \quad \mathbf{H}_2 = [1 \quad 0]$$

където

$T$  – период на обзора на РЛС.

Грешката на филтъра за съпровождане за единична цел (BC) се дефинира като разлика между вектора на измерванията и оценката на вектора на състоянието на BC [1], [2], т.е.

$$(3) \quad \zeta(k+1) = \mathbf{Z}(k+1) - \mathbf{H} \hat{\mathbf{X}}(k+1)$$

Точността на формираната от филтъра оценка влияе в голяма степен върху вземането на решението за коректно асоцииране на данните към наличните траектории на BC, чрез което оказва влияние върху нивото на сигурност при УВД и върху безопасността на полетите. Точността на оценката, осигурявана от филтъра за съпровождане на траектории с BPNN зависи от избора на най-подходящата невронна архитектура, от метода на формиране на вектора на признаците, от точността на обучаващия алгоритъм и от точността и сходимостта на филтъра, осигурявани от метода за инициализация на BPNN. В настоящата статия е представено експериментално сравнение на седем метода за инициализация на

BPNN, използвани в алгоритъм за филтрация с предсказване при съпровождане на траектории на ВС, с цел въз основа на анализ на получените резултати от тях да бъде избран методът, който осигурява най-малки първоначални грешки при съпровождане на траекториите, добро качество на обучението и най-бърза сходимост на BPNNF. Тъй като част от съществуващите методи за инициализация на BPNN [3]–[5] са подходящи само при решаване на класификационни проблеми, за експерименталното сравнение са избрани само методи, подходящи за използване както при класификация и разпознаване, така и за филтрация с предсказване. Някои от тях са сравнително нови методи [4]. Резултатите от изпълнение на BPNNF за всеки метод са сравнени с резултатите, получени със стандартен рекурсивен филтър на Калман (Kalman Filter, KF) чрез експеримент от 50 симулации по метода Monte Carlo (MC) в програмната среда *MATLAB*. Използвани са симулирани входни данни в съответствие с кинематичния модел на неманевриращо ВС (1) – (2) в полярна координатна система и записи с реални радиолокационни данни от монопулсен вторичен обзорен радар за УВД CMSSR–401 [6]. Изследването е фокусирано върху получените грешки на BPNNF при инициализацията на алгоритъма и бързината на сходимост на обучението му.

## 2. Методи за инициализация на BPNN

В този раздел на статията са представени основните уравнения на седем от методите за инициализация на BPNN за филтрация с предсказване, участващи в сравнението.

*Метод 1:* Обичайният подход за инициализация на НМ [4], [5], включен като база за сравнение. При него първоначалните стойности на синаптичните тегла се избират като случайни числа, равномерно разпределени в интервала  $[-0,05; 0,05]$  или в  $[-1; 1]$ . Тези два варианта за краткост ще бъдат означени с “*a*” и “*b*”.

*Метод 2:* Метод на Russo [4], [5] – инициализация, при която теглата се избират като случайни числа, разпределени равномерно в  $\left[-\frac{2,4}{F_i}; \frac{2,4}{F_i}\right]$ , където с  $F_i$  е означен общият брой синаптични връзки, свързващи неврон  $i$  от даден слой на НМ с невроните от предходния слой.

*Метод 3:* Случайна инициализация, при която стойностите на теглата са ограничени отдолу, като [4]

$$(4) \quad \sqrt{\frac{\mu}{F_i}} < |w_{ij}|$$

където

$\mu$  – скорост на обучение на BPNN.

Подходяща горна граница в случая е 0,1 [4].

*Метод 4:* Метод на Shimodaira [4], който може да бъде формулиран в следните стъпки:

*Ст. 1.* Изчисляване на параметъра

$$(5) \quad b = \left| f^{-1}(1 - \varepsilon) - f^{-1}(\varepsilon) \right|$$

където

$f$  – активиращата функция на неврона (по подразбиране сигмоидална);  
 $\varepsilon = 0,1$ .

Ст. 2. Изчисляване на

$$(6) \quad \hat{w} = \frac{b}{\sqrt{2} k^* F_i}$$

Ст. 3. Генериране на случайно число  $\alpha_{ij}$  в интервала

$$(7) \quad -\gamma^* \leq \alpha_{ij} \leq \gamma^*$$

Ст. 4. Изчисляване на синаптичните тегла  $w_{ij}$ , свързващи неврон  $i$  в даден слой с неврон  $j$  от предходния слой при инициализацията като

$$(8) \quad w_{ij} = \hat{w} \sqrt{\alpha_{ij} + 1}$$

след което Ст. 2 и Ст. 3 се повтарят  $F_i$  пъти.

Ст. 5. Праговата стойност  $w_{i0}$  се избира равна на нула.

Като стойности на параметрите  $k^*$  и  $\gamma^*$  се препоръчват комбинациите (2; 0,3), (5; 0,3), (8; 0,3), (2; 0,6), (5; 0,6), (8; 0,6), (2; 0,9), (5; 0,9) и (8; 0,9), означени последователно от “a” до “i” [4].

*Метод 5:* Метод SCAWI (Statistically Controlled Activation Weight Initialization) [4], базиран на концепцията на т. нар. парализиран процент неврони. Тази концепция се дефинира чрез тестване колко пъти даден неврон е в състояние на насищане и величината на най-малко един изходен сигнал е висока. Тогава синаптичните тегла на невроните в скрития и в изходния слой при инициализацията се определят като

$$(9) \quad w_{ij}^{input} = \frac{1,3}{\sqrt{1 + n_{inp} v^2}} r_{ij}$$

$$(10) \quad w_{ij}^{hid} = \frac{1,3}{\sqrt{1 + n_{hid}}} r_{ij}$$

където

$n_{inp}$ ,  $n_{hid}$  – брой неврони съответно в скрития и в изходния слой;

$v$  – средноквадратичната стойност на входните сигнали на невроните;

$r_{ij}$  – случайни числа, равномерно разпределени в  $[-1; 1]$ .

*Метод 6:* Метод на Nguyen–Widrow, при който стойността на параметъра  $\beta$  [5] се определя като

$$(11) \quad \beta = 0,7(n_{hid})^{F_i}$$

при което стойностите на всички синаптични тегловни коефициенти  $w_{ij}^{(init)}$  се избират като случайни числа с равномерно разпределение в интервала  $[-1; 1]$ , след което стойностите на теглата на невроните в скрития слой се модифицират като

$$(12) \quad w_{ij}^{(0)} = \frac{\beta}{\|\mathbf{W}^{(init)}\|} w_{ij}^{(init)}$$

където

$$\mathbf{W}^{(init)} = \left[ w_{11}^{(init)}, \dots, w_{n_{inp} n_{hid}}^{(init)} \right].$$

*Метод 7:* Алгоритъм на Nguyen–Widrow–Russo, който се формулира в следните стъпки [5]:

*Ст. 1.* Нулиране на сумирация параметър за  $\|\mathbf{W}\|$  и изчисляване на параметъра  $\beta$ , т. е.

$$(13) \quad w(0) = 0; \quad \beta(0) = 0,7e^{-\frac{\ln(n_{hid})}{F_i}}$$

*Ст. 2.* Разпределяне на всички тегла равномерно в  $[-1; 1]$  и следващо модифициране на теглата в изходния слой като

$$(14) \quad w_{ij}^{out}(1) = \left( \frac{2,4}{F_i} \right) w_{ij}^{out}(0)$$

след което

$$(15) \quad w(k+1) = w(k) + w_{ij}^2(k)$$

$$(16) \quad w = \sqrt{w(k)}; \quad \beta(k) = \frac{\beta(k-1)}{w}$$

*Ст. 3.* Изчисляване на теглата в скрития слой като

$$(17) \quad w_{ij}^{hid}(k+1) = \beta w_{ij}^{hid}(k)$$

### 3. Архитектура на BPNNF, входни данни и обучение

Архитектурата на BPNNF включва многослойна нерекурсивна невронна мрежа със закъснителни звена на входа за филтрация с предсказване една стъпка напред във времето и ред на предсказване  $p = 3$ , с пълни вътрешни връзки между невроните. BPNN има 3 входа, един скрит слой и 3 изхода. Активиращите функции на невроните са сигмоидални за входния и за скрития слой и линейни за изходния слой. Тази архитектура е избрана за оптимална по евристичен път в отделен експеримент.

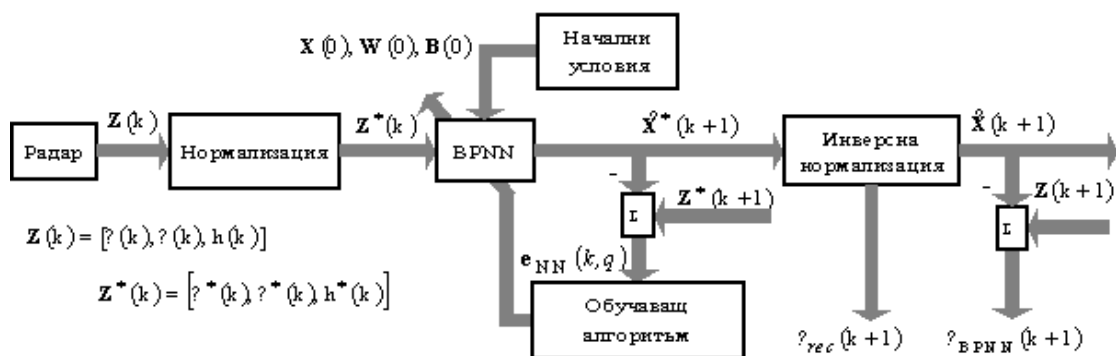
Входни данни за BPNNF са полярните координати разстояние радар-цел  $r$ , NM; азимут  $\theta$ , rad и височина  $h$ , feet за всяко от съпровождащите ВС, които формират  $\mathbf{Z}(k)$  за всеки обзор  $k$ , след предварителна обработка нормализационна процедура, която осигурява нулева средна стойност и дисперсия единица на еле-

ментите на входния вектор за BPNN за подобряване на сходимостта на обучаващия алгоритъм. Входният вектор в нормализиран вид се формулира като [7]

$$(18) \quad \mathbf{P}_n = [\mathbf{Z}^*(k-1), \mathbf{Z}^*(k-2), \dots, \mathbf{Z}^*(k-p)]$$

където

$\mathbf{Z}^*(k-1)$  – вектор на нормализираните радарни измервания за всички ВС, с който се формира обучаващата извадка с размерност  $Q$  ( $Q$  на брой ВС).



Фиг. 1. Схема на филтрация с предказване една стъпка напред чрез BPNN

Блоковата схема на обработката на данните с BPNN в случая е показана на Фиг. 1. Оптималният ред на предказване  $p = 3$  е избран в отделен експеримент. За обучение на BPNNF е използван стандартният алгоритъм за гъвкаво разпространение на грешката (Resilient Propagation, RP) [5], [7], [8]. Обучението продължава до достигане на глобалния минимум на функцията на грешката след обработване на цялата обучаваща извадка данни

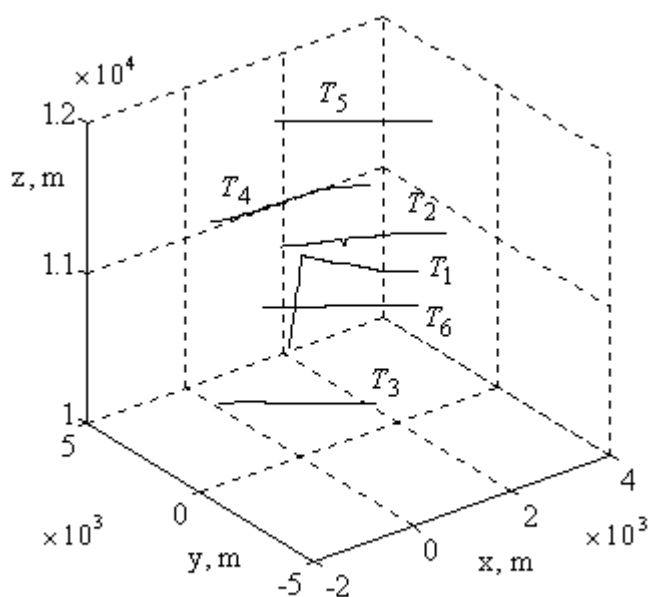
$$(19) \quad E = \frac{1}{Q} \sum_{q=1}^Q \xi_q = \frac{1}{2} \sum_{q=1}^Q \sum_{j=1}^{n_{out}} (d_{jq} - y_{jq})^2$$

или е достигнат максималният брой итерации на обучението. Променливите  $d_{jq}$  и  $y_{jq}$  от (19) са желаният изходен сигнал и изходният сигнал на неврон  $j$  в изходния слой за обучаващ пример  $q$ . След обработката на данните с BPNN, оригиналните дименсии на величините се възстановяват чрез инверсна нормализационна процедура. Обработката на дадена траектория с BPNNF продължава до удовлетворяване на критерий за прекратяване изчислението на траекторията липса на данни за ВС от 6 последователни периода на сканиране на радара [9].

#### 4. Резултати от експериментите

За компютърното моделиране на разработения алгоритъм (BPNNF) са използвани симулирани и реални входни данни за 6 траектории на неманевриращи ВС. Броят траектории за паралелна обработка с BPNNF е избран с оглед наличния хардуер за извършване на изчисленията, т.е.  $Q = 6$ . Използваните реални входни данни са записани от изхода на плот екстрактора на монопулсен втори-чен обзорен радар за УВД СМSSR-401 [6] с период на сканиране  $T=10$  s. Реалните траектории, показани на Фиг. 2 в Декартови координати, са с дължина от 140 последователни обзора на РЛС. Тези траектории са използвани като прототипи за моделиране на траекториите за МС симулациите. Стандартните

отклонения на грешките на РЛС при измерване на координатите са 0,05 NM за  $\rho$ ;



Фиг. 2. Реални траектории на ВС, използвани при експериментите

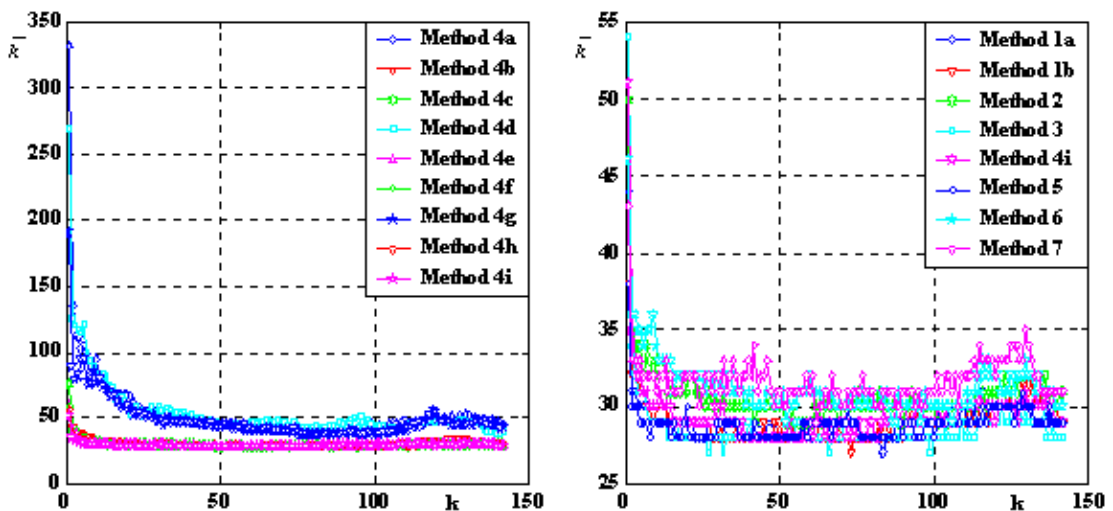
0,07° за  $\theta$  и 100 feet за  $h$  [6]. Те са използвани при моделиране на  $\mathbf{v}(k)$  в съответствие с (1)–(2). Интегралните функции на разпределение на вероятностите на елементите на  $\mathbf{v}(k)$  са верифицирани с  $\chi^2$ -тест с ниво на значимост  $\alpha = 0,05$ . Резултатите от изпълнение на BPNNF са сравнени с тези, получени от стандартен рекурсивен KF при стандартно отклонение на ускорението на ВС  $2g \text{ m/s}^2$  [1]. Ковариационната матрица на KF при инициализацията е  $\mathbf{P}_0 = 1/\epsilon \mathbf{I}$ ,

при  $\epsilon = 0,001$  [10]. Параметри за сравнението при MC симулациите са среден брой итерации на обучението на BPNNF  $\bar{k}$ , осреднените за 50 MC симулации стойности на функцията на грешката в края на обучението  $\bar{E}_{fin}$  и на квадратен корен от средноквадратичните грешки (Root Mean Square Error, RMSE)  $\zeta$  на филтрите при съпровождане на траекториите. Параметри на изчислителната сложност на алгоритмите са средно процесорно време  $\bar{t}_{CPU, s}$  и среден брой операции с плаваща запетая (Float Operations Per Second, FLOPS)  $\bar{n}_{FLOPS}$  за обработка на данните от един обзор за шестте траектории. Всички експериментални резултати са получени при използване на персонален компютър с Intel Celeron 500 PPGA и SDRAM 128 MB. Резултатите от MC симулациите на BPNNF за различните методи на инициализация са представени в Таблица 1. От деветте варианта на метода, най-близо до оптималния случай се оказва *Метод 4i*, който се сравнява с останалите шест разгледани метода за определяне на най-подходящия метод за инициализация на BPNNF. Графичните зависимости на параметрите на обучението и на грешките при съпровождане на траектория  $T_1$  за двете групи сравнения са показани съответно на Фиг. 3 и Фиг. 4. Грешките за останалите траектории не са показани, тъй като характерът на изменението им е аналогичен на този при  $T_1$ . Вижда се, че при всички изследвани методи за инициализация има сходност на обучението, като BPNN се обучава най-бързо при инициализация по *Метод 1* и *Метод 3 – 5* (средно за 29 итерации). Зависимостите на средната грешка в края на обучението при всички методи в този експеримент имат приблизително един и същи характер на изменение с течение на времето, а средните й стойности са близки една до друга. От двата варианта на инициализация по *Метод 1*, вариант *a*, при който синаптичните тегла се

инициализират в по-тесен затворен интервал, се оказва по-близо до оптималния случай в сравнение с вариант *b*.

Таблица 1. Резултати от МС симулациите

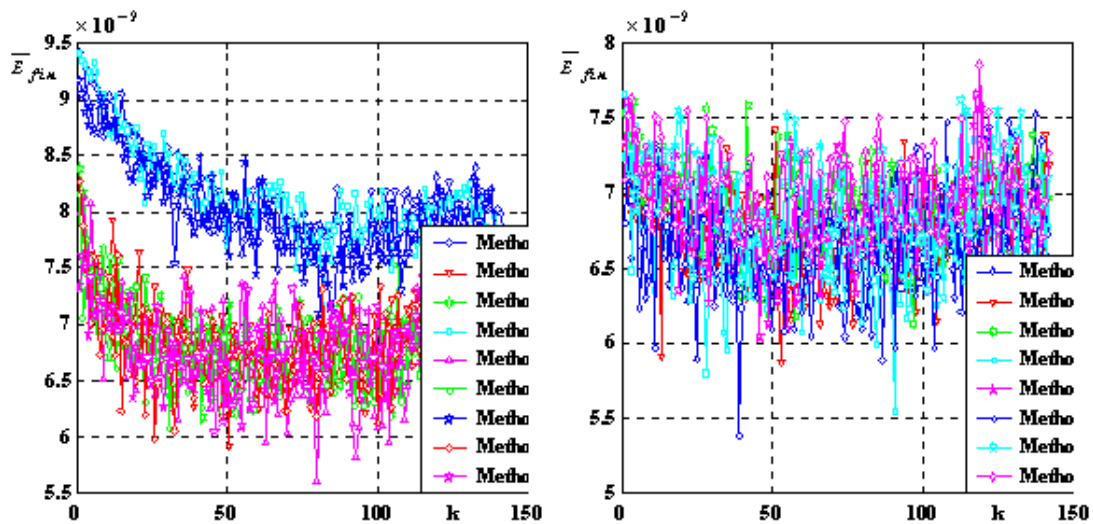
Метод	$\bar{k}^T$	$\bar{E}_{f_{\text{min}}} \times 10^{-6}$	$\bar{\gamma}_\gamma, \text{NM}$		$\bar{\gamma}_\gamma, \text{mrad}$		$\bar{\gamma}_h, \text{ft}$		$\bar{t}_{\text{CPU}}, \text{s}$	$\bar{n}_{\text{FLOPS}}$
			$T_1$	$T_2$	$T_1$	$T_2$	$T_1$	$T_2$		
1a	29	0,668	0,18	0,14	8,3	3,5	4,93	4,88	1,12	20914
1b	29	0,673	0,22	0,19	10,5	11,9	14,92	17,01	1,08	21077
2	31	0,693	0,21	0,19	10,3	9,2	18,83	18,78	1,13	22096
3	29	0,665	0,19	0,14	9,0	3,8	3,61	4,34	1,11	20686
4a	53	0,815	1,17	1,12	56,1	51,0	82,63	82,02	1,56	37449
4b	31	0,688	0,54	0,48	25,9	20,2	27,57	25,98	1,16	22277
4c	30	0,684	0,38	0,33	18,1	12,7	13,39	12,26	1,12	21616
4d	54	0,816	1,15	1,10	55,0	49,7	80,68	79,76	1,74	37681
4e	30	0,680	0,54	0,47	27,5	20,0	27,73	26,05	1,18	21544
4f	30	0,682	0,38	0,32	17,8	12,5	12,87	11,74	1,11	21654
4g	49	0,798	1,12	1,05	53,0	47,1	76,79	74,60	1,54	34356
4h	30	0,682	0,52	0,46	24,5	18,8	24,87	23,16	1,11	21569
4i	29	0,679	0,37	0,31	17,4	12,0	11,91	10,81	1,10	21212
5	29	0,668	0,17	0,13	7,5	4,1	8,58	8,53	1,17	20848
6	31	0,695	0,25	0,24	13,5	11,3	16,98	17,28	1,15	22440
7	32	0,701	0,21	0,19	11,3	9,5	18,98	18,85	1,18	22741
KF			1,10	1,14	17,1	12,4	129,69	129,67	0,02	2718



Фиг. 3. Среден брой итерации на обучението на BPNMF за различните методи и а инициализация



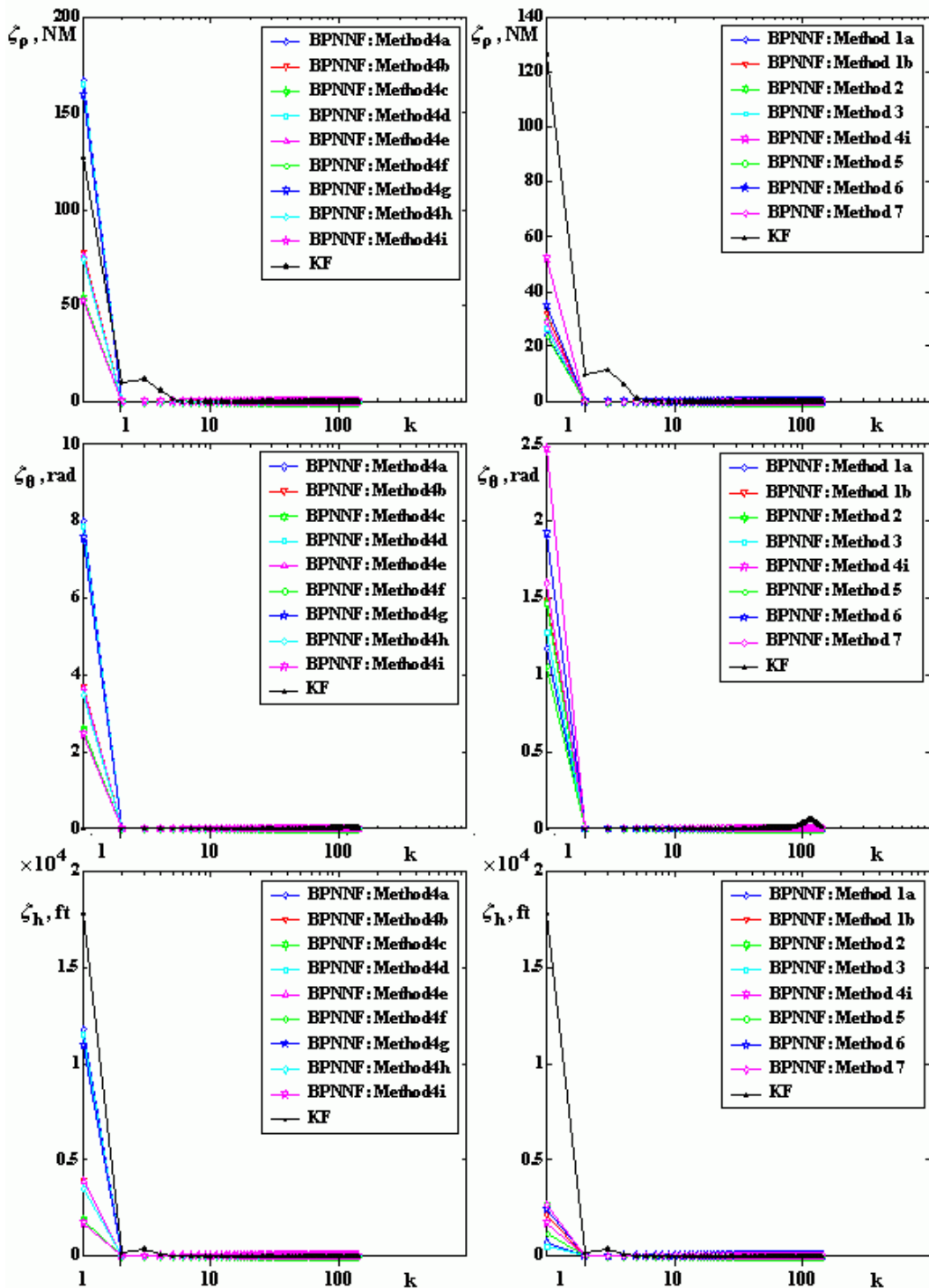
При последния са необходими и повече изчислителни ресурси. При него и стойностите на средните грешки (RMSE) са относително ниски, тъй като филтрираните позиции попадат сравнително близо до екстраполираните вследствие малките тегловни коефициенти.



Фиг. 4. Средни грешки в края на обучението на BPNNF за различните методи на инициализация

При инициализация по метода на Russo (*Метод 2*) също се получават малки грешки при съпровождане, но в този случай са необходими повече итерации на обучение на мрежата, респ. повече процесорно време и брой FLOPS. При използване на *Метод 3* грешката на обучението на BPNNF е най-близо до глобалния минимум, който се достига средно за най-малко време, изискват се най-малко FLOPS в сравнение с останалите методи и се получават сравнително най-малки средни грешки при съпровождане на траекториите. При използване на метода SCAWI (*Метод 5*) също се получават малки средни грешки при съпровождане, но се наблюдава относително забавена сходимост на обучението (необходимо е повече процесорно време) без значително нарастване на броя FLOPS. При инициализацията на BPNNF по метода на Nguyen-Widrow и неговата разновидност – алгоритъма на Nguyen-Widrow-Russo (*Метод 6 – 7*) теглата се разпределят с по-малка прецизност в пространството на признаците [5], [8], което забавя сходимостта на обучението (изискват се повече итерации и изчислителни ресурси) и глобалният минимум на функцията на грешката се апроксимира с понижена точност, а стойностите на грешките при съпровождане на траекториите при инициализацията на мрежата са леко повишени. Все пак, поради по-ниските стойности на теглата в изходния слой на BPNN при алгоритъма на Nguyen-Widrow-Russo се наблюдава подобрене на точността при апроксимиране на минимума на грешката и по-малки стойности на първоначалните грешки при съпровождане в сравнение с метода на Nguyen-Widrow, поради което и средните грешки са по-малки. При него обаче сходимостта на обучението е най-забавена в сравнение с останалите изследвани методи и се изискват най-много итерации на обучение и изчислителни ресурси. При инициализацията по метода на Shimodaira качеството на обучение е прибли-

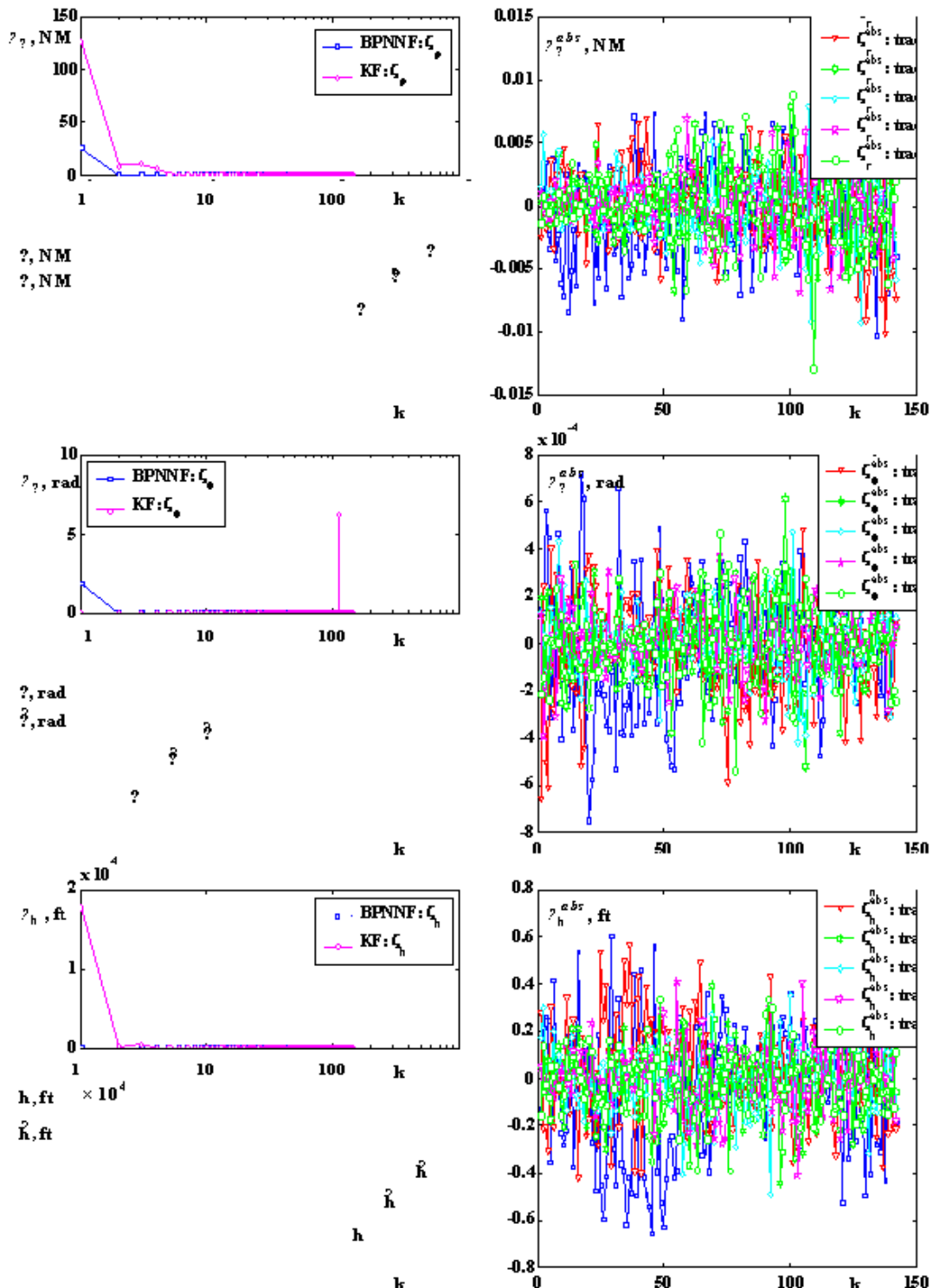
зително еднакво с това при *Метод 1а*, обаче средното време за обучение на BPNN е най-голямо спрямо това на другите методи.



Фиг. 5. Средни грешки (RMSE) при събръщане на траектории с BPNN за различните методи на инициализация, сравнени с KF

Това се дължи на специфичния характер на инициализацията. При този метод се наблюдава известна неравномерност в разпределението на теглата, което води до получаване на сравнително малки първоначални грешки за едни траектории и доста по-големи грешки за други. Резултатите от симулациите показват, че най-

близо до оптималната инициализация на BPNNF се намира *Метод 3*, при който мрежата се обучава най-бързо и се получават сравнително най-малки грешки при обучението и при филтрацията с предсказване.



Фиг. 6. Грешки (RMSE), действителни и екстраполирани траектории с BPNNF и с KF (реални входни данни) за траектория  $T_1$

Фиг. 7. Абсолютни грешки на BPNNF в установен режим за шестте реални траектории

Резултатите от едно изпълнение на BPNNF за този случай на инициализация при използване на записи реални радиолокационни данни за същите траектории са

представени в Таблица 2. Грешките при съпровождане (RMSE), действителните и екстраполираните траектории за траектория  $T_1$  са показани на Фиг. 6.

Таблица 2. Резултати от изпълнение на BPNNF (Метод 3) при реални входни данни

$\bar{k}^T$	$\bar{E}_{fin} \times 10^{-8}$	$\bar{\zeta}_\rho, \text{NM}$		$\bar{\zeta}_\theta, \text{mrad}$		$\bar{\zeta}_h, \text{ft}$		$\bar{t}_{\text{CPU}}, \text{s}$	$\bar{n}_{\text{FLOPS}}$
		$T_1$	$T_2$	$T_1$	$T_2$	$T_1$	$T_2$		
33	0,724	0,19	0,14	13,3	18,1	3,50	3,44	1,38	23294

Анализът на сходимост на BPNNF показва, че най-големи стойности на грешките при съпровождане се наблюдават при инициализацията на филтъра, като след това бързо намаляват и в установен режим (от обзор  $k = 3$  нататък) са почти постоянни, а осреднените им стойности са близки до получените при MC симулациите. Грешките на BPNNF по отношение на  $\rho$  и  $h$  при инициализацията са по-малки от съответните грешки на KF, поради което осреднените им стойности върху цялата дължина на траекториите са също по-малки. Грешките при инициализацията на BPNNF спрямо  $\theta$  са по-големи от тези на KF, поради специфичния подход на инициализацията на филтъра. Зависимостите на абсолютните грешки при съпровождане  $\zeta^{abs}$  в установен режим представляват почти еднотипни стохастични процеси (Фиг. 7). Техните емпирични разпределения на вероятностите се схождат с нормално разпределение с доверителна вероятност над 90 %, получена като краен резултат от серия статистически тестове ( $\chi^2$  – тест, тест на Shapiro-Wilks, тест на Колмогоров-Смирнов и други емпирични тестове), извършени с помощта на програмния пакет *STATGRAPHICS*<sup>®</sup> [11]. Грешките при съпровождане на траекториите с BPNNF в установен режим не надвишават максимално допустимите грешки при неманевриращи ВС, регламентирани в [2] и [9], а именно 0,2 NM за  $\rho$ ; 1,5° за  $\theta$  и 100/16 ft за  $h$ .

## ЗАКЛЮЧЕНИЕ

На база на направения сравнителен анализ на работата на BPNNF при различните методи за инициализация беше избран метода, който осигурява най-бърза сходимост на обучаващия алгоритъм и най-висока точност при съпровождане на траекториите. Резултатите от експериментите показват, че в този случай BPNNF работи ефективно и с по-висока точност в сравнение с рекурсивния KF. Освен това областта на установен режим на BPNNF започва по-рано, отколкото при KF, което заедно с постигането на по-високата точност при формиране на екстраполираните позиции на ВС е предпоставка за подобряване качеството на асоцииране на данните, чрез което се повишават степента на сигурност и надеждност на автоматизираните системи за УВД и безопасността на полетите.

## ЛИТЕРАТУРА

- [1] BLACKMAN S. Multiple Target Tracking with Radar Applications. Norwood, Artech House, 1986.

- [2] BAR-SHALOM Y. (editor). Multiple-Target Tracking, **1** (1990), **2** (1992), Dedham, Artech House.
- [3] KOSKO B. Neural Networks and Fuzzy Systems. Prentice Hall International, 1992.
- [4] HERNÁNDEZ–ESPINOSA C., M. FERNÁNDEZ–REDONDO. Multilayer Feedforward Weight Initialization, Int. Joint Conf. on Neural Networks, Washington DC, July 15–19 2001, **1**, 166–170.
- [5] LOONEY C. G., Pattern Recognition Using Neural Networks. New York, Oxford University Press, 1997.
- [6] Monopulse Secondary Surveillance Radar System Description. Technical Report, Cardion Inc., Report no. 131-162A.
- [7] CICHOCKI A., R. UNBEHAUEN. Neural Networks for Optimization and Signal Processing., John Wiley & Sons & B. G. Teubner, Stuttgart, 1993.
- [8] Neural Networks Toolbox User's Guide Version 3.0., The Math Works, Inc., 1992–1998.
- [9] International Standards and Recommended Practices: Aeronautical Telecommunications, Annex 10 to the Convention of International Civil Aviation., **4** (Surveillance Radar and Collision Avoidance Systems), Attachment A, 2nd Edition – July 1998, ICAO, 185–186.
- [10] ТАРАСОВ В. Г. Обработка информации в автоматизированных системах управления. изд. ВВИА им. проф. Н. Е. Жуковского, Москва, 1974.
- [11] STATGRAPHICS Version 4.0 Plus For Windows. User's Manual. Magnugistics Inc. USA, 1995, 631.

## **COMPARISON OF SEVEN WEIGHT INITIALIZATION METHODS FOR BPNN TRACKING FILTER**

**Mimi Daneva**

*Department of Radiocommunications, Technical University of Sofia  
8 Kliment Ohridski str., 1000 Sofia, BULGARIA*

**Keywords:** *Radar Data Processing, Back-Propagation Neural Networks.*

**Summary:** *In this paper an experimental comparison among seven different methods for weight initialization of a BPNN filter for tracking on non-maneuvering aircrafts is presented. The performances of the BPNN tracking filter using different weight initialization are evaluated and compared with standard recursive Kalman filter. A Monte Carlo verification of the results is done. An illustrative example using recorded real radar database is shown.*