Enhanced random search based incremental extreme learning machine

Guang-Bin Huanga, Lei Chen

Neurocomputing 71 (2008) 3460–3468

Introducción

- Este artículo presenta una mejora al incremental-ELM propuesto por los mismos autores previamente.
- El método incremental añade nodos ocultos aleatoriamente.
- Algunos de esos nodos afectan muy poco a la reducción del error residual y por tanto pueden eliminarse.

Single-hidden-layer feedforward networks (SLFNs)

$$f_n(\mathbf{x}) = \sum_{i=1}^{n} \beta_i g_i(\mathbf{x}) = \sum_{i=1}^{n} \beta_i G(\mathbf{x}, \mathbf{a}_i, b_i), \quad \mathbf{a}_i \in \mathbf{C}^d,$$

$\mathbf{x} \in \mathbf{C}^d$, $b_i \in C$, $\beta_i \in C$,

(1) Additive hidden nodes:

$$g_i(\mathbf{x}) = g(\mathbf{a}_i \cdot \mathbf{x} + b_i), \quad \mathbf{a}_i \in \mathbf{R}^d, \ b_i \in R,$$

where g is the activation function of hidden nodes.

(2) RBF hidden nodes:

$$g_i(\mathbf{x}) = g(b_i \|\mathbf{x} - \mathbf{a}_i\|), \quad \mathbf{a}_i \in \mathbf{R}^d, \ b_i \in R.$$

(3) Fully complex hidden nodes [15]:

$$g_i(\mathbf{x}) = \prod_{l=1}^{s_i} g(\mathbf{a}_{il} \cdot \mathbf{x} + b_i), \quad \mathbf{a}_{il} \in \mathbf{C}^d, \quad \mathbf{x} \in \mathbf{C}^d, \quad b_i \in C,$$

$$\beta_i \in C, \tag{4}$$

where s_i is an integer constant.

I-ELM

I-ELM [8] randomly adds nodes to the hidden layer one by one and freezes the output weights of the existing hidden nodes when a new hidden node is added. I-ELM [8] is a "*random search*" method in the sense that in theory the residual error of I-ELM will decrease and I-ELM moves toward the target function further whenever a hidden node is randomly added.

EI-ELM

Let $L^2(X)$ be a space of functions f in a measurable compact subset X of the d-dimensional space $\mathbf{C}^d$ such that $|f|^2$ are integrable. For $u, v \in L^2(X)$, the inner product $\langle u, v \rangle$ is defined by $\langle u, v \rangle = \int_X u(\mathbf{x})\overline{v(\mathbf{x})}\,\mathrm{d}\mathbf{x}$. The closeness between network function f_n and the target function f is measured by the L^2 distance:

$$\|f_n - f\| = \left[\int_X (f_n(\mathbf{x}) - f(\mathbf{x}))\overline{(f_n(\mathbf{x}) - f(\mathbf{x}))}\,\mathrm{d}\mathbf{x}\right]^{1/2} \quad (5)$$

residual error of f_n as $e_n \equiv f - f_n$ where $f \in L^2(X)$

Teoremas

Theorem 2.1. *Given a SLFN with any nonconstant piecewise continuous hidden nodes* $G(\mathbf{x}, \mathbf{a}, b)$, *if* $span\{G(\mathbf{x}, \mathbf{a}, b) : (\mathbf{a}, b) \in \mathbf{C}^d \times C\}$ *is dense in* L^2, *then for any continuous target function* f *and any function sequence* $\{g_n(\mathbf{x}) = G(\mathbf{x}, \mathbf{a}_n, b_n)\}$ *randomly generated based on any continuous sampling distribution,* $\lim_{n\to\infty} \|f - (f_{n-1} + \beta_n g_n)\| = 0$ *holds with probability one if*

$$\beta_n = \frac{\langle e_{n-1}, g_n \rangle}{\|g_n\|^2}. \tag{7}$$

Theorem 2.2. *Given an SLFN with any nonconstant piecewise continuous hidden nodes* $G(\mathbf{x}, \mathbf{a}, b)$, *if* $span\{G(\mathbf{x}, \mathbf{a}, b) : (\mathbf{a}, b) \in \mathbf{C}^d \times C\}$ *is dense in* L^2, *for any continuous target function* f *and any randomly generated function sequence* $\{g_n\}$ *and any positive integer* k, $\lim_{n\to\infty} \|f - f_n^*\| = 0$ *holds with probability one if*

$$\beta_n^* = \frac{\langle e_{n-1}^*, g_n^* \rangle}{\|g_n^*\|^2}, \tag{12}$$

where $f_n^* = \sum_{i=1}^n \beta_i^* g_i^*$, $e_n^* = f - f_n^*$ *and* $g_n^* = \{g_i | \min_{(n-1)k+1 \leqslant i \leqslant nk} \|(f - f_{n-1}^*) - \beta_n g_i\|\}$.

Algoritmo

step 1 **Initialization:** Let $L = 0$ and residual error $E = t$, where $t = [t_1, \ldots, t_N]^{\mathrm{T}}$.

step 2 **Learning step:**

while $L < L_{\max}$ and $\|E\| > \varepsilon$

(a) Increase by 1 the number of hidden nodes L: $L = L + 1$.

(b) **for** $i = 1 : k$

(i) Assign random parameters $(\mathbf{a}_{(i)}, b_{(i)})$ for the new hidden node L according to any continuous sampling distribution probability.

(ii) Calculate the output weight $\beta_{(i)}$ for the new hidden node:

$$\beta_{(i)} = \frac{E \cdot H_{(i)}^T}{H_{(i)} \cdot H_{(i)}^T}. \tag{16}$$

(iii) Calculate the residual error after adding the new hidden node L:

$$E_{(i)} = E - \beta_{(i)} \cdot H_{(i)}. \tag{17}$$

endfor

(c) Let $i^* = \{i | \min_{1 \leqslant i \leqslant k} \|E_{(i)}\|\}$. Set $E = E_{(i)}$, $\mathbf{a}_L = \mathbf{a}_{(i^*)}$, $b_L = b_{(i^*)}$, and $\beta_L = \beta_{(i^*)}$.

endwhile

Table 2
Performance comparison between EI-ELM and I-ELM (both with 200 sigmoid hidden nodes)

Name	EI-ELM (sigmoid, $k = 10$)			I-ELM (sigmoid, $k = 1$)		
	Mean	Dev.	Time (s)	Mean	Dev.	Time (s)
Abalone	**0.0818**	0.0020	2.5801	0.0920	0.0046	0.2214
Ailerons	**0.0558**	0.0024	9.5017	0.1023	0.0353	0.7547
Airplane	**0.0804**	0.0039	0.5669	0.1016	0.0093	0.0499
Auto price	**0.0896**	0.0022	0.3141	0.0977	0.0069	0.0329
Bank	**0.0631**	0.0031	3.9838	0.1173	0.0068	0.3237
Boston	**0.1055**	0.0098	0.4332	0.1167	0.0112	0.0515
California	**0.1494**	0.0015	9.3336	0.1683	0.0049	0.5448
Census (8L)	**0.0829**	0.0012	11.476	0.0923	0.0023	0.8667
Computer activity	**0.0941**	0.0028	3.7511	0.1201	0.0125	0.2794
Delta ailerons	**0.0445**	0.0062	2.7735	0.0525	0.0078	0.2620
Delta elevators	**0.0582**	0.0032	3.7971	0.0740	0.0126	0.2708
Kinematics	<u>0.1393</u>	0.0028	3.4373	<u>0.1418</u>	0.0033	0.2810
Machine CPU	<u>0.0466</u>	0.0060	0.3112	<u>0.0504</u>	0.0079	0.0234
Puma	<u>0.1840</u>	0.0017	3.9531	<u>0.1861</u>	0.0041	0.3236
Pyrim	**0.1414**	0.0341	0.3062	0.1867	0.0628	0.0374
Servo	**0.1518**	0.0116	0.3235	0.1662	0.0124	0.0218

Table 3
Performance comparison between EI-ELM and I-ELM (both with 200 RBF hidden nodes)

Name	EI-ELM (RBF, $k = 10$)			I-ELM (RBF, $k = 1$)		
	Mean	Dev.	Time (s)	Mean	Dev.	Time (s)
Abalone	**0.0829**	0.0027	5.6006	0.0938	0.0053	0.5030
Ailerons	**0.0774**	0.0129	36.016	0.1430	0.0298	3.2769
Airplane	**0.0633**	0.0057	0.9578	0.0992	0.0166	0.0751
Auto price	**0.1139**	0.0189	0.4031	0.1261	0.0255	0.0468
Bank	**0.0730**	0.0022	9.8079	0.1157	0.0097	0.7782
Boston	**0.1077**	0.0084	0.7972	0.1320	0.0126	0.0657
California	**0.1503**	0.0022	17.133	0.1731	0.0081	1.3656
Census (8L)	**0.0810**	0.0016	19.922	0.0922	0.0029	1.7928
Computer activity	**0.1153**	0.0021	10.092	0.1552	0.0282	0.8220
Delta ailerons	**0.0448**	0.0065	4.6169	0.0632	0.0116	0.4327
Delta elevators	**0.0575**	0.0047	7.3541	0.0790	0.0123	0.6321
Kinematics	**0.1213**	0.0017	8.3114	0.1555	0.0122	0.6953
Machine CPU	**0.0554**	0.0148	0.4114	0.0674	0.0177	0.0447
Puma	**0.1752**	0.0022	9.7983	0.1913	0.0180	0.7872
Pyrim	**0.1209**	0.0431	0.4423	0.2241	0.1752	0.0434
Servo	**0.1379**	0.0151	0.4031	0.1524	0.0200	0.0391

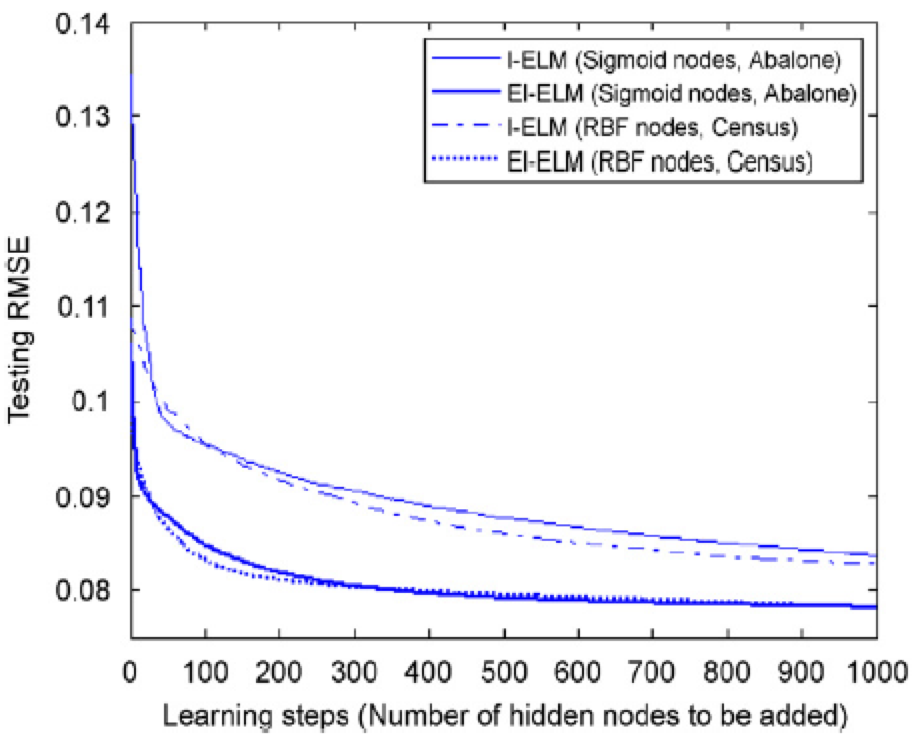

Fig. 1. The testing error updating curves of EI-ELM and I-ELM.

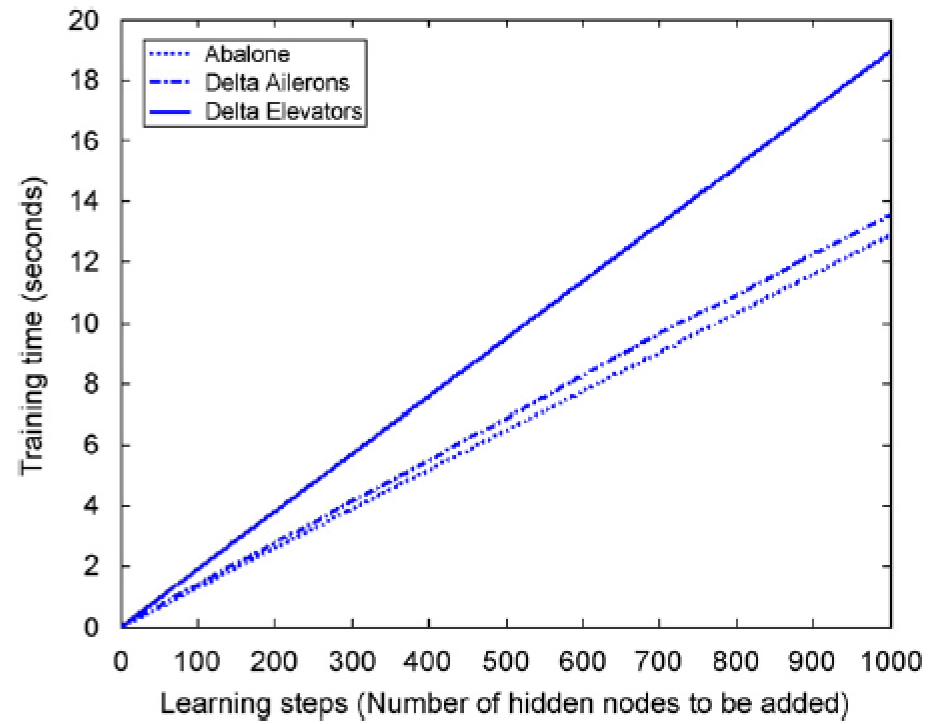

Fig. 2. Training time spent by EI-ELM is linearly increased with the number of hidden nodes to be added.

Table 4
Performance comparison between EI-ELM with 50 Sigmoid hidden nodes and I-ELM with 500 sigmoid hidden nodes

Problems	EI-ELM (50 sigmoid hidden nodes)						I-ELM (500 sigmoid hidden nodes, $k = 1$)		
	$k = 10$			$k = 20$					
	Mean	Dev.	Time (s)	Mean	Dev.	Time (s)	Mean	Dev.	Time (s)
Abalone	0.0878	0.0033	0.6506	0.0876	0.0015	1.5785	0.0876	0.0033	0.7695
Ailerons	0.0640	0.0066	2.3766	**0.0571**	0.0022	6.2519	0.0824	0.0232	1.8810
Airplane	0.0922	0.0061	0.1389	0.0862	0.0040	0.2921	0.0898	0.0067	0.1466
Auto price	0.0924	0.0112	0.0814	**0.0897**	0.0104	0.1658	0.0948	0.0158	0.0561
Bank	0.1066	0.0058	0.9965	0.0896	0.0036	3.1058	**0.0757**	0.0032	0.7914
Boston	0.1133	0.0101	0.1065	0.1102	0.0061	0.2232	0.1084	0.0096	0.1033
California	0.1591	0.0034	2.2423	0.1548	0.0033	4.9486	0.1543	0.0019	1.5665
Census (8L)	0.0899	0.0017	2.8655	0.0865	0.0011	6.1100	0.0871	0.0018	2.1199
Computer activity	0.1075	0.0057	0.9342	**0.0991**	0.0036	2.3311	0.1057	0.0078	0.7185
Delta ailerons	0.0474	0.0062	0.7006	0.0467	0.0042	1.4570	0.0468	0.0052	0.6340
Delta elevators	0.0615	0.0049	0.9502	**0.0586**	0.0038	2.5385	0.0640	0.0055	0.6516
Kinematics	0.1420	0.0029	0.8655	0.1416	0.0019	2.9017	0.1406	0.0014	0.7117
Machine CPU	0.0498	0.0155	0.0750	0.0467	0.0148	0.1577	0.0474	0.0040	0.0645
Puma	0.1846	0.0018	0.9856	0.1827	0.0017	2.7264	0.1856	0.0039	0.7983
Pyrim	0.1514	0.0419	0.0782	**0.1300**	0.0405	0.1533	0.1712	0.0626	0.0810
Servo	0.1634	0.0129	0.0795	0.1558	0.0121	0.1611	0.1589	0.0124	0.0642

Table 5
Performance comparison between EI-ELM with 50 RBF hidden nodes and I-ELM with 500 RBF hidden nodes

Problems	EI-ELM (50 RBF hidden nodes)						I-ELM (500 RBF hidden nodes, $k = 1$)		
	$k = 10$			$k = 20$					
	Mean	Dev.	Time (s)	Mean	Dev.	Time (s)	Mean	Dev.	Time (s)
Abalone	0.0907	0.0034	1.4036	0.0871	0.0023	3.0006	0.0872	0.0022	1.2121
Ailerons	0.0973	0.0229	9.0306	**0.0775**	0.0033	19.071	0.1129	0.0295	8.1818
Airplane	0.0943	0.0168	0.2347	0.0813	0.0102	0.5487	0.0772	0.0082	0.1940
Auto price	0.1187	0.0159	0.0998	**0.1104**	0.0148	0.2110	0.1231	0.0133	0.1189
Bank	0.0989	0.0031	2.4460	0.0888	0.0023	5.4199	0.0843	0.0058	1.9382
Boston	0.1197	0.0107	0.1845	**0.1171**	0.0078	0.3621	0.1214	0.0103	0.1872
California	0.1624	0.0049	4.2339	0.1579	0.0027	8.8326	0.1582	0.0027	3.8482
Census (8L)	0.0864	0.0026	4.9858	0.0846	0.0020	11.796	0.0860	0.0018	4.8536
Computer activity	0.1295	0.0068	2.4905	**0.1201**	0.0024	5.5878	0.1358	0.0177	2.1267
Delta ailerons	0.0469	0.0067	1.1800	**0.0466**	0.0039	2.4763	0.0544	0.0076	1.0361
Delta elevators	0.0603	0.0049	1.8515	**0.0602**	0.0039	4.2506	0.0685	0.0099	1.5399
Kinematics	0.1346	0.0025	2.0913	**0.1306**	0.0019	4.6727	0.1425	0.0095	1.7042
Machine CPU	0.0622	0.0281	0.1067	**0.0511**	0.0114	0.2031	0.0614	0.0274	0.0875
Puma	0.1789	0.0020	2.4465	**0.1770**	0.0012	5.2821	0.1850	0.0119	1.9709
Pyrim	0.1214	0.0345	0.1016	**0.0989**	0.0286	0.2079	0.2179	0.1545	0.1071
Servo	0.1487	0.0133	0.0985	0.1434	0.0120	0.1958	0.1410	0.0151	0.0982

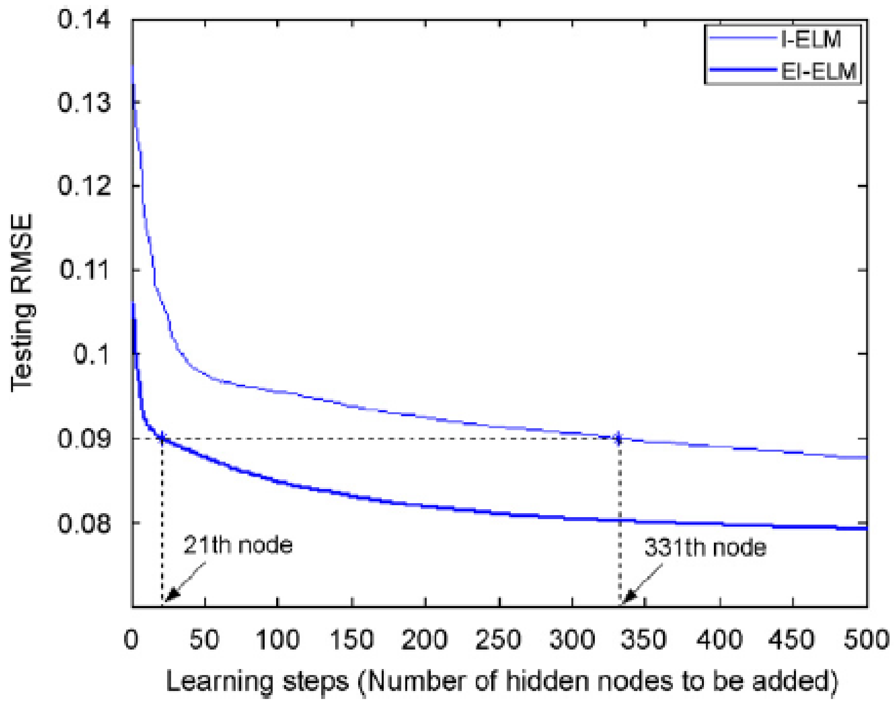

Fig. 3. Testing RMSE performance comparison between EI-ELM and I-ELM (with sigmoid hidden nodes) for Abalone case.

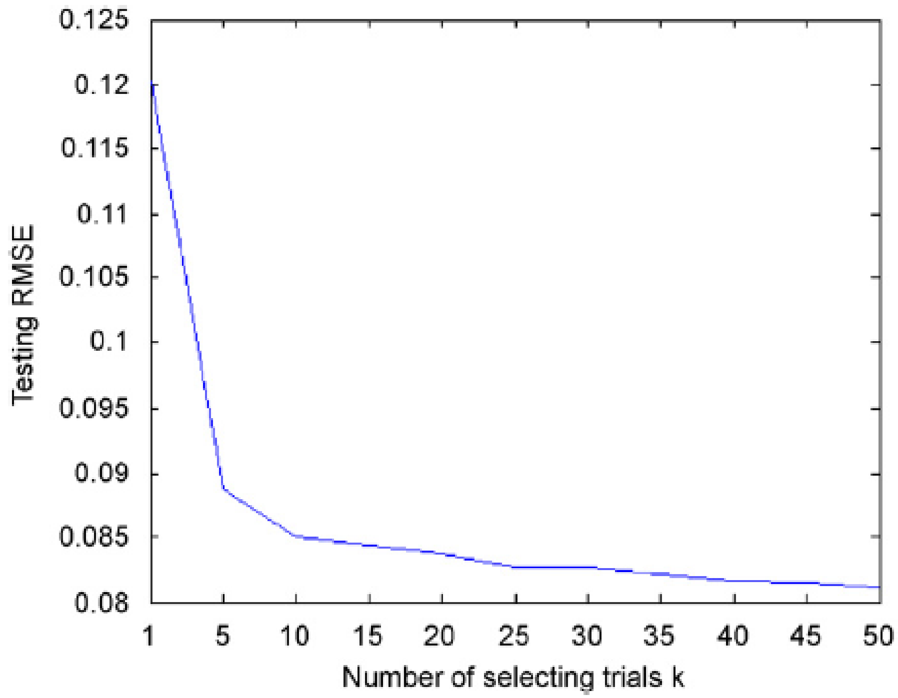

Fig. 4. Effect of number of selecting trials k on the generalization performance of EI-ELM in airplane case.

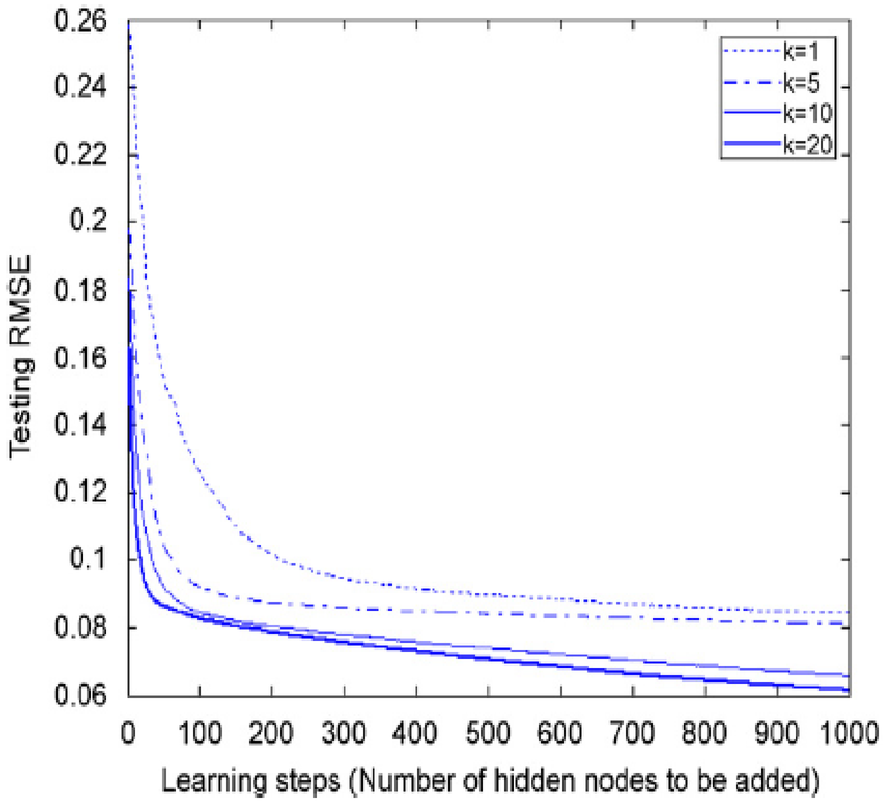

Fig. 6. Testing RMSE updating progress with new hidden nodes added and different number of selecting trials k in airplane case.

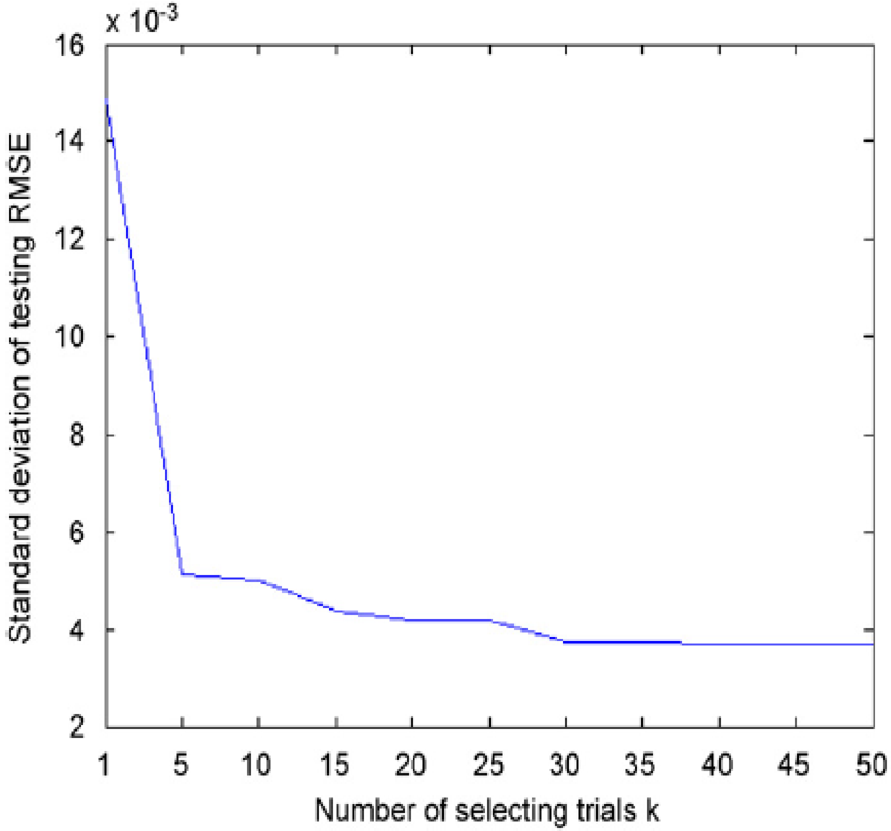

Fig. 5. Effect of number of selecting trials k on the stability of the generalization performance of EI-ELM in airplane case.