

# Теория и методы цифровой обработки сигналов

## ПАРАЛЛЕЛЬНЫЕ ВЫЧИСЛЕНИЯ В ЛЕСТНИЧНЫХ RLS-АЛГОРИТМАХ

Джиган В.И.

Государственное унитарное предприятие г. Москвы Научно-производственный центр «Электронные вычислительно-информационные системы» (ГУП НПП «ЭЛВИС») а/я 19, Центральный проспект, Зеленоград, г. Москва К-460, Россия 124460  
Тел.: +7-095-531-1961. Эл. почта: [djigan@elvees.ru](mailto:djigan@elvees.ru). Интернет: <http://www.elvees.ru>

**Аннотация:** Рассматривается применение параллельных вычислений в быстрых лестничных RLS-алгоритмах адаптивной фильтрации. В алгоритмах используется скользящее окно при оценке корреляционной матрицы адаптивного фильтра и (или) динамическая регуляризация этой матрицы. Приведены вычислительные процедуры алгоритмов и структуры адаптивных фильтров на их основе. Такие фильтры предназначены для обработки нестационарных сигналов в приложениях, не требующих вычисления весовых коэффициентов адаптивного фильтра в явном виде.

В настоящее время адаптивные фильтры широко используются в радиоэлектронных устройствах систем связи, радиолокации и бытовой техники [1]. Обычно, в основе работы большинства адаптивных фильтров находится простейший градиентный алгоритм по критерию наименьшего среднеквадратического отклонения или его модификации. Такие алгоритмы имеют наименьшую арифметическую сложность среди алгоритмов адаптивной фильтрации:  $2N$  арифметических операций на одну итерацию, где  $N$  – число весовых коэффициентов фильтра. Кроме того, они характеризуются алгоритмической простотой – несложным математическим описанием. Однако градиентные алгоритмы обладают рядом известных недостатков, которые заметно проявляются при обработке нестационарных сигналов.

Рекурсивные алгоритмы по критерию наименьших квадратов (Recursive Least Squares, RLS) [2] являются одними из наиболее эффективных алгоритмов адаптивной фильтрации. Они обеспечивают лучшие показатели качества (меньшую длительность переходного процесса и меньшие значения ошибок в установившемся режиме) по сравнению с градиентными алгоритмами. Однако сложность таких алгоритмов равна  $O(N^2)$  арифметических операций на одну итерацию или  $O(N)$  операций в вычислительно эффективных (быстрых) версиях. Несмотря на это, RLS-алгоритмы также применяются в адаптивных фильтрах, поскольку современные высокопроизводительные цифровые сигнальные процессоры уже позволяют реализовывать сложные алгоритмы.

При обработке нестационарных сигналов отмеченную арифметическую сложность RLS-алгоритмов приходится увеличивать примерно в два раза за счет применения скользящего окна (для обеспечения следящих свойств) при оценке корреляционной матрицы входного сигнала адаптивного фильтра, регуляризации этой матрицы (для обеспечения ее стабильного обращения) или в четыре раза при одновременном использовании обоих приемов. В случае использования двух или четырех процессоров, вычислительную нагрузку на один процессор в таких алгоритмах можно уменьшить, соответственно в два или четыре раза, при построении RLS-алгоритмов на основе параллельных вычислений [3].

Существуют также быстрые RLS-алгоритмы, в которых внутри каждой рекурсивной по времени итерации содержится  $N$  итераций, вычисляемых при изменении порядка адаптивного фильтра от  $n = 1$  до  $n = N$ . Эти фильтры называются лестничными [4]. В лестничных фильтрах применение скользящего окна известно лишь для нескольких разновидностей адаптивных алгоритмов. Применение регуляризации или совместное применение регуляризации и скользящего окна для таких алгоритмов в литературе не встречалось.

В настоящей работе рассматривается применение скользящего окна и (или) регуляризации в лестничных RLS-алгоритмах, базирующихся на параллельных вычислениях. Данные алгоритмы при реализации в арифметике с плавающей точкой и при соответствующей инициализации математически эквивалентны другим RLS-алгоритмам и друг другу. Рассматриваются алгоритмы для одноканальных адаптивных фильтров с комплексными весовыми коэффициентами  $h_n(k)$ , рис. 1. Здесь  $x(k)$  – входной,  $d(k)$  – требуемый и  $y(k)$  – выходной сигналы,  $\alpha(k)$  – сигнал ошибок,  $k$  – индекс дискретного времени. Алгоритмы для фильтров с действительными весовыми коэффициентами являются частными случаями рассматриваемых алгоритмов при замене всех комплексных переменных и арифметических операций на действительные переменные и операции.

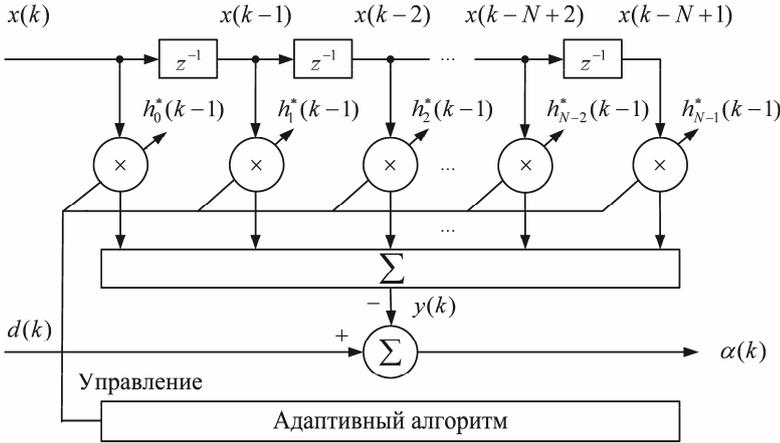


Рис. 1. Адаптивный фильтр

Особенностью лестничных адаптивных фильтров является отсутствие в них вычисления весовых коэффициентов в явном виде. Интересным выходным сигналом таких фильтров является сигнал ошибки. Примерами использования адаптивных фильтров, в которых интересным сигналом является лишь сигнал ошибки  $\alpha(k) = d(k) - y(k)$ , могут служить задачи подавления сигналов электрического и акустического эха. Существуют два основных типа лестничных адаптивных фильтров,

в процессе работы которых вычисляются априори ошибки  $\alpha(k) = d(k) - \mathbf{h}_N^H(k-1)\mathbf{x}_N(k)$  и апостериори ошибки  $e(k) = d(k) - \mathbf{h}_N^H(k)\mathbf{x}_N(k)$ . Здесь  $\mathbf{h}_N(k)$  и  $\mathbf{x}_N(k)$  – векторы весовых коэффициентов и входного сигнала.

Используя приемы, аналогичные [3, 4], можно получить ряд параллельных лестничных RLS-алгоритмов. При этом необходимо сформировать векторы  $\mathbf{x}_F(k) = [\mu^{0.5}x(k-L), x(k), \mu^{0.5}\xi p(k-L), \xi p(k)]$ ,  $\mathbf{d}_F(k) = [\mu^{0.5}d(k-L), d(k), 0, 0]$  и матрицу  $\mathbf{S}_F = \text{diag}(-1, 1, -1, 1)$ . Здесь  $L$  – число отсчетов, определяемое длиной скользящего окна. Параметр  $\mu = \lambda^L$ , где  $0 \ll \lambda \leq 1$  – параметр экспоненциального взвешивания,  $\xi p(k)$  – сигнал динамической регуляризации [2]. Варианты алгоритмов (с бесконечным окном, со скользящим окном, с бесконечным окном и регуляризацией, со скользящим окном и регуляризацией), определяются наличием соответствующих двух или четырех компонент в указанных векторах при  $F = 1$ ,  $F = 2$  или  $F = 4$ . Векторы ошибок линейного предсказания  $\mathbf{a}_F^{f(n)}(k)$ ,  $\mathbf{a}_F^{b(n)}(k)$ ,  $\mathbf{e}_F^{f(n)}(k)$ ,  $\mathbf{e}_F^{b(n)}(k)$  и моделирования  $\mathbf{a}_F^{(n)}(k)$ ,  $\mathbf{e}_F^{(n)}(k)$  также состоят из одной, двух или четырех компонент.

Следуя рассмотренным выше приемам, можно получить параллельный лестничный алгоритм, который приведен в табл. 1. Соответствующие алгоритму структуры вычисления ошибок показаны на рис. 2 и рис. 3. В качестве интересующего выходного сигнала адаптивного фильтра (см. рис. 1) используется сигнал  $\alpha(k) = \alpha_x^{(N+1)}(k)$ , где  $\alpha_x^{(N+1)}(k)$  – элемент вектора  $\mathbf{a}_F^{(N+1)}(k)$ , зависящий от  $x(k)$ . В уравнениях алгоритма (табл. 1)  $\Gamma^{f(n)}(k)$ ,  $\Gamma^{b(n)}(k)$  и  $\gamma^{(n)}(k)$  – коэффициенты отражения, а  $E^{f(n)}(k)$  и  $E^{b(n)}(k)$  – энергии ошибок линейного предсказания. Верхний индекс  $(n)$  служит для обозначения вычислений, связанных с изменением порядка фильтра как  $n = 1, \dots, N$ , а верхние индексы  $f$  и  $b$  означают вычисления, относящиеся к линейному предсказанию вперед и назад (forward prediction и backward prediction).

Рассмотренный алгоритм может быть модифицирован за счет использования соотношений между априори или апостериори ошибками в параллельных RLS-алгоритмах [3]:  $\mathbf{e}_F^{f(n)}(k) = \mathbf{a}_F^{f(n)}(k)\Phi_F^{(n)}(k-1)$ ,  $\mathbf{e}_F^{b(n)}(k) = \mathbf{a}_F^{b(n)}(k)\Phi_F^{(n)}(k)$ ,  $\mathbf{a}_F^{f(n)}(k) = \mathbf{e}_F^{f(n)}(k)[\Phi_F^{(n)}(k-1)]^{-1}$ ,  $\mathbf{a}_F^{b(n)}(k) = \mathbf{e}_F^{b(n)}(k)[\Phi_F^{(n)}(k)]^{-1}$ ,  $\mathbf{a}_F^{(n)}(k) = \mathbf{e}_F^{(n)}(k)[\Phi_F^{(n)}(k)]^{-1}$  и  $\Phi_F^{(n+1)}(k) = \Phi_F^{(n)}(k) - \mathbf{e}_F^{b(n)H}(k)\mathbf{e}_F^{b(n)}(k)/E^{b(n)}(k)$ ,  $\Phi_F^{(n+1)}(k) = \Phi_F^{(n)}(k)[\mathbf{I}_F - \mathbf{a}_F^{b(n)H}(k)\mathbf{a}_F^{b(n)}(k)\Phi_F^{(n)}(k)/E^{b(n)}(k)]$ , где  $\mathbf{I}_F$  – единичная матрица, а  $\Phi_F^{(n)}(k)$  – матрица отношений правдоподобия.

С помощью упомянутых приемов получено семь разновидностей параллельных лестничных алгоритмов: на основе априори и апостериори ошибок без обратных связей и с обратными связями, на основе только априори или только апостериори ошибок без обратных связей и с обратными связями, а также алгоритм на основе использования QR-разложения без операций извлечения квадратного корня. Детали одноименных алгоритмов с бесконечным окном могут быть найдены в работе [4].

Таблица 1. Лестничный RLS-алгоритм на основе априори и апостериори ошибок

| Вычисления  | Ссылки  |
|---|---|
| <b>Initialization :</b> $E^{f(n)}(0) = \delta^2, E^{b(n)}(0) = \delta^2 \lambda^{n-1}, \Delta^{(n)}(0) = 0, \kappa^{(n)}(0) = 0,$<br>$\mathbf{\alpha}_F^{b(n)}(0) = \mathbf{0}_F, \mathbf{e}_F^{b(n)}(0) = \mathbf{0}_F, \Gamma^{f(n)}(0) = 0, \Gamma^{b(n)}(0) = 0, \gamma^{(n)}(0) = 0, n = 1, 2, \dots, N$   | (1.0)   |
| <b>For</b> $k = 1, 2, \dots, K$<br>$\mathbf{\alpha}_F^{f(1)}(k) = \mathbf{x}_F(k), \mathbf{\alpha}_F^{b(1)}(k) = \mathbf{x}_F(k), \mathbf{e}_F^{f(1)}(k) = \mathbf{x}_F(k)\mathbf{S}_F, \mathbf{e}_F^{b(1)}(k) = \mathbf{x}_F(k)\mathbf{S}_F,$<br>$\mathbf{\alpha}_F^{(1)}(k) = \mathbf{d}_F(k), \mathbf{e}_F^{(1)}(k) = \mathbf{d}_F(k)\mathbf{S}_F$<br><b>For</b> $n = 1, 2, \dots, N$<br>$E^{f(n)}(k) = \lambda E^{f(n)}(k-1) + \mathbf{e}_F^{f(n)}(k)\mathbf{\alpha}_F^{f(n)H}(k)$<br>$E^{b(n)}(k) = \lambda E^{b(n)}(k-1) + \mathbf{e}_F^{b(n)}(k)\mathbf{\alpha}_F^{b(n)H}(k)$<br>$\Delta^{(n)}(k) = \lambda \Delta^{(n)}(k-1) + \mathbf{e}_F^{b(n)}(k-1)\mathbf{\alpha}_F^{f(n)H}(k)$<br>$\kappa^{(n)}(k) = \lambda \kappa^{(n)}(k-1) + \mathbf{\alpha}_F^{b(n)}(k)\mathbf{e}_F^{(n)H}(k)$<br>$\mathbf{\alpha}_F^{f(n+1)}(k) = \mathbf{\alpha}_F^{f(n)}(k) + \Gamma^{f(n)*}(k-1)\mathbf{\alpha}_F^{b(n)}(k-1)$<br>$\mathbf{\alpha}_F^{b(n+1)}(k) = \mathbf{\alpha}_F^{b(n)}(k-1) + \Gamma^{b(n)*}(k-1)\mathbf{\alpha}_F^{f(n)}(k)$<br>$\mathbf{\alpha}_F^{(n+1)}(k) = \mathbf{\alpha}_F^{(n)}(k) + \gamma^{(n)*}(k-1)\mathbf{\alpha}_F^{b(n)}(k)$<br>$\Gamma^{f(n)}(k) = -\Delta^{(n)}(k)/E^{b(n)}(k-1)$<br>$\Gamma^{b(n)}(k) = -\Delta^{(n)*}(k)/E^{f(n)}(k)$<br>$\gamma^{(n)}(k) = -\kappa^{(n)}(k)/E^{b(n)}(k)$<br>$\mathbf{e}_F^{f(n+1)}(k) = \mathbf{e}_F^{f(n)}(k) + \Gamma^{f(n)*}(k)\mathbf{e}_F^{b(n)}(k-1)$<br>$\mathbf{e}_F^{b(n+1)}(k) = \mathbf{e}_F^{b(n)}(k-1) + \Gamma^{b(n)*}(k)\mathbf{e}_F^{f(n)}(k)$<br>$\mathbf{e}_F^{(n+1)}(k) = \mathbf{e}_F^{(n)}(k) + \gamma^{(n)*}(k)\mathbf{e}_F^{b(n)}(k)$<br><b>End for</b> $n$<br><b>End for</b> $k$ | (1.1)<br>(1.2)<br>(1.3)<br>(1.4)<br>(1.5)<br>(1.6)<br>(1.7)<br>(1.8)<br>(1.9)<br>(1.10)<br>(1.11)<br>(1.12)<br>(1.13)<br>(1.14) |

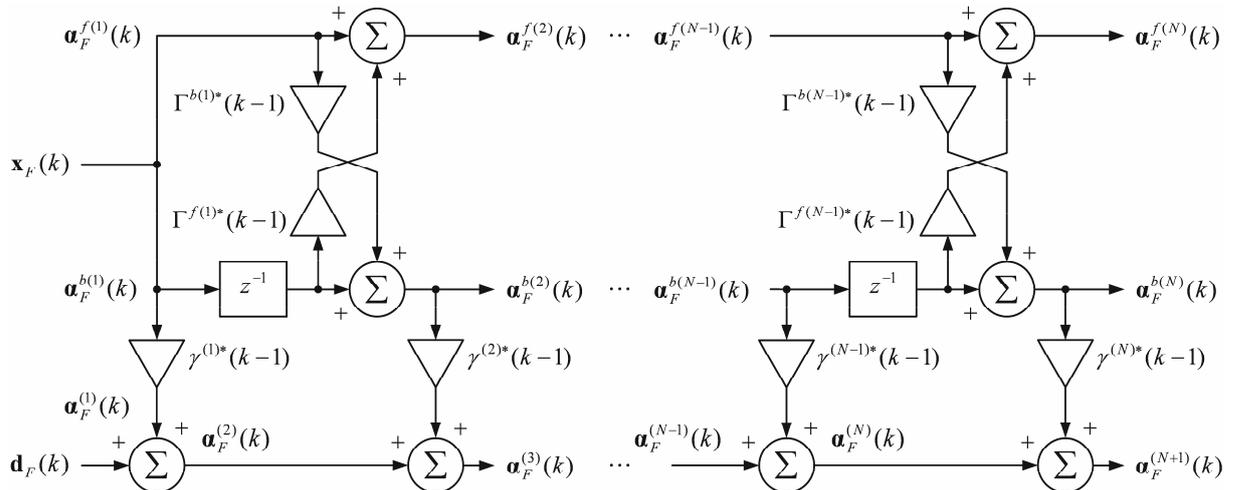


Рис. 2. Параллельный лестничный адаптивный фильтр на основе априори ошибок

Оценка арифметической сложности алгоритма (табл. 1) примерно в  $F$  раз больше такой оценки одноименного алгоритма с бесконечным окном, сложность которого равна  $10N$  сложений,  $14N$  умножений и  $3N$  делений. Такое увеличение сложности в случае применения скользящего окна и (или) регуляризации имеет место в представленных в настоящей работе модификациях параллельных лестничных алгоритмов и в алгоритмах [3]. В случае применения параллельных вычислений, нагрузку на один процессор удастся, как

уже отмечалось, уменьшить в  $F$  раз, поскольку все вычисления, связанные с элементами вектора  $\mathbf{x}_F(k)$ , в параллельных алгоритмах независимы друг от друга.

В [2] была продемонстрирована эффективность применения скользящего окна и регуляризации корреляционной матрицы адаптивного при обработке нестационарных сигналов. Аналогичные результаты справедливы и в случае применения лестничных фильтров, поскольку рассмотренные в работе алгоритмы и соответствующие им одноканальные варианты алгоритмов [2] математически эквивалентны.

Модульная структура лестничных фильтров (рис. 2 и рис. 3) обеспечивает их эффективную программную или аппаратную реализацию. Рассмотренные лестничные RLS-алгоритмы могут применяться в приложениях, где интересующим выходным сигналом адаптивного фильтра является лишь сигнал ошибки моделирования. Представленные в работе алгоритмы – это результат дальнейшего развития прикладной библиотеки алгоритмов и программ [5].

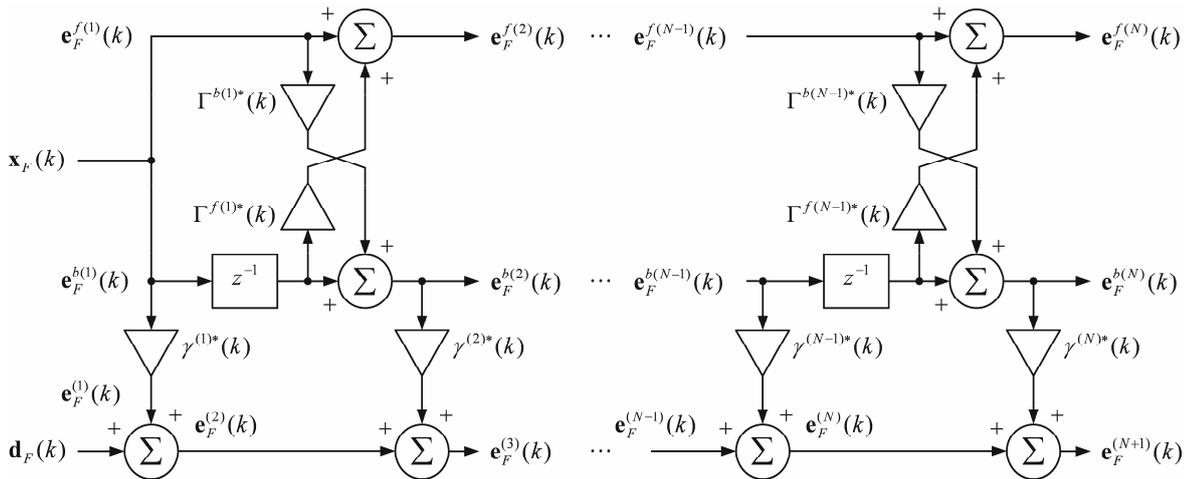


Рис. 3. Параллельный лестничный адаптивный фильтр на основе апостериори ошибок

**Литература**

1. Benesty J., Huang Y. (Eds). Adaptive signal processing: applications to real-world problems. – Berlin, Heidelberg, New York, Springer-Verlag, 2003. – 356 p.
2. Джиган В.И. Многоканальные RLS- и быстрые RLS-алгоритмы адаптивной фильтрации // Успехи современной радиоэлектроники. – 2004. – №11. – С. 48–77.
3. Djigan V.I. RLS adaptive filtering algorithms based on parallel computations // Radioengineering: Proceedings of Czech and Slovak Technical Universities and URSI Committers. – 2005. – Vol. 14. – №3. – P. 28–36.
4. Джиган В.И. Многообразие лестничных RLS-алгоритмов адаптивной фильтрации // Цифровая обработка сигналов. – 2005. – №3. – С. 2–12.
5. Джиган В.И. Библиотека алгоритмов адаптивной фильтрации // Доклады 6-й Международной конференции «Цифровая обработка сигналов и ее применения (DSPA-2004)». – Москва, 2004. – Том 1. – С. 89–94.

**ON PARALLEL COMPUTATIONS IN LATTICE RLS ALGORITHMS**

Djigan V.

Electronic VLSI Engineering & Embedded Systems (ELVEES) R&D Center of Microelectronics  
 POB 19, Centralny Prospect, Zelenograd, Moscow K-460, Russia 124460  
 Tel.: +7-095-531-1961. E-mail: [djigan@elvees.ru](mailto:djigan@elvees.ru). URL: <http://www.elvees.ru>

Today, adaptive filters are intensively used as the portions of equipment in communication systems, radars and consumer electronics. The examples are electrical and acoustic echo cancellers, communication and acoustic channel equalizers, noise cancellers, adaptive arrays etc. Generally, the adaptive filters use the computationally simplest gradient least mean squares criterion algorithm or the different versions of the algorithm. However, the simplest adaptive filtering algorithms are characterized by a number of known performance drawbacks that become visible in case of non-stationary signal processing.

Recursive Least Squares (RLS) algorithms [1] are among of the most effective adaptive filtering algorithms. The algorithms provide better performance namely shorter transient response and smaller errors in comparing with gradient algorithms. The arithmetic complexity of RLS algorithms exceeds the complexity of gradient algorithms. Despite of that, RLS algorithms are also used in adaptive filters because the modern digital signal processors have enough resources for the implementation of complex algorithms.

In case of non-stationary signal processing the RLS algorithms complexity has to be doubled due to the application of a sliding window or regularization in correlation matrix estimation. In sliding window and regularization case the algorithms complexity is increased four times in comparing with the growing window RLS algorithms. If two or four processors are used to implement the modified algorithms, the computational load per processor is decreased two or four times if RLS algorithms are built on the base of parallel computation principles [2].

The given paper presents the application of the sliding window or (and) regularization in lattice RLS algorithms [3]. The new algorithms development is based on the mentioned parallel computations. Following the papers [2, 3], seven sorts of new parallel lattice algorithms were developed. They are a priori and a posteriori error algorithms with and without feedbacks, only a priori or only a posteriori error algorithms with and without feedbacks, and square root free QR decomposition algorithms. Full computational procedure of a priori and a posteriori error algorithm without feedbacks and the architecture of an adaptive filter for the error computation are presented in the paper. Evaluation of the algorithm complexity is presented as well.

The modular architecture of the new parallel lattice algorithms assures effective software or hardware implementation. The algorithms can be used in the applications of adaptive filters, where only modeling error signal is used as a desired output signal. The examples of the applications are electrical and acoustic echo cancellation.

The algorithms, presented in the paper, are the result of the permanent development of applied library in adaptive signal processing [4]. Presently the library consists of about 300 different adaptive filtering algorithms, some of which were described in [1–3].

#### References

1. Djigan V.I. Multichannel RLS and fast RLS adaptive filtering algorithms // *Successes of Modern Radioelectronics*. – 2004. – №11. – P. 48–77.
2. Djigan V.I. RLS adaptive filtering algorithms based on parallel computations // *Radioengineering: Proceedings of Czech and Slovak Technical Universities and URSI Committers*. – 2005. – Vol. 14. – №3. – P. 28–36.
3. Djigan V.I. Lattice RLS adaptive filtering algorithms diversity // *Digital Signal Processing*. – 2005. – №3. – P. 2–12.
4. Djigan V.I. Library of adaptive filtering algorithms // *Proceedings of the 6-th International Conference on Digital Signal Processing and its Applications (DSPA-2004)*. – Moscow, 2004. – Vol. 1. – P. 88–94.

