Adaptive Regularization Matrix for Affine Projection Algorithm
Young-Seok Choi, Hyun-Chool Shin, Member, IEEE, and Woo-Jin Song, Member, IEEE

Abstract—We propose an adaptive regularized affine projection algorithm (AR-APA) with a regularization matrix. Instead of the conventional fixed and weighted identity matrix for regularization, the proposed AR-APA incorporates a nonidentity regularization matrix which is dynamically updated. The adaptation of the regularization matrix is accomplished by the normalized stochastic gradient of mean-square error. Consequently, an efficient and robust APA is derived. Experimental results comparing the proposed AR-APA to the conventional algorithms clearly indicate its superior performance.

Index Terms—Adaptive regularization matrix, affine projection algorithms (APAs).

I. INTRODUCTION

THE normalized least mean square (NLMS) is one of widely used adaptive algorithms due to its simplicity and ease of implementation. However, its convergence rate is significantly reduced for correlated input signals [1]–[3]. To overcome this problem, the affine projection algorithm (APA) was proposed by Ozeki and Umeda [4]. While the LMS-type filters update the weights based only on the current input vectors, the APA updates the weights on the basis of the last K input vectors [4]–[7]. In the APA, the inversion of the rank-deficient matrix may give rise to the singularity. To avoid this situation, a positive constant $\delta$ called the regularization parameter is used [1], [2]. We use the regularized APA (R-APA) as opposed to the simple APA in order to highlight the presence of the regularization parameter $\delta$; APA is reserved for the case $\delta = 0$. It is also known that the regularization parameter plays a critical role in the convergence performance of the R-APA [8]. In the R-APA, the regularization parameter $\delta$ governs both the rate of convergence and the misadjustment error. To meet the conflicting requirements of fast convergence and low misadjustment error, the regularization parameter needs to be adjusted and optimized.

An earlier work for the NLMS with a time-varying regularization parameter has been presented [9]. In this paper, the optimal regularization matrix which is an extension of a scalar regularization parameter is introduced for the R-APA.

With this in mind, we propose an adaptive R-APA (AR-APA) with a regularization matrix. Conventional R-APA uses the fixed and weighted identity matrix for regularization. The proposed AR-APA incorporates a nonidentity regularization matrix which is dynamically adjusted. To make the proposed algorithm robust, we introduce the normalized gradient of the mean-square error into the update for the regularization matrix. As a result, an efficient and robust algorithm is derived. We show that the proposed algorithm has little additional complexity compared to the conventional R-APA. Through experiments, we illustrate that the proposed algorithm has better performance than the conventional R-APA and is comparable to the recursive least square (RLS) algorithm in terms of the convergence rate and the misadjustment error.

This brief is organized as follows. In Section II, we introduce the conventional R-APA and develop the AR-APA. In addition, we provide the stability issue of the proposed algorithm. Section III illustrates the experimental results and Section IV presents conclusions.

II. AR-APA

Consider data $d(\hat{i})$ that arise from the system identification model

$$d(\hat{i}) = u_i e^o + v(\hat{i})$$  \hspace{1cm} (1)

where $e^o$ is a column vector for the impulse response of an unknown system that we wish to estimate, $v(\hat{i})$ accounts for measurement noise and $u_i$ denotes the $1 \times M$ input vector

$$u_i = [u(\hat{i}) u(\hat{i} - 1) \cdots u(\hat{i} - M + 1)]$$  \hspace{1cm} (2)

A. Conventional Regularization in R-APA

Let $w_i$ be an estimate for $e^o$ at iteration $\hat{i}$. The R-APA computes $w_i$ via

$$w_i = w_{i-1} + \mu U_i U_i^* (U_i U_i^* + \delta I)^{-1} e_i$$  \hspace{1cm} (3)

where

$$U_i = \begin{bmatrix} u_i \\ u_{i-1} \\ \vdots \\ u_{i-K+1} \end{bmatrix}, \quad d_i = \begin{bmatrix} d(\hat{i}) \\ d(\hat{i} - 1) \\ \vdots \\ d(\hat{i} - K + 1) \end{bmatrix},$$

$$e_i = d_i - U_i w_{i-1}, \quad \mu$$ is the step-size, $\delta$ is the regularization parameter, and $^*$ denotes the Hermitian transpose. The regularization parameter is not employed only to avoid the inversion of the rank-deficient matrix $U_i U_i^*$ but also to play a critical role in the convergence performance of the R-APA. A small regularization parameter will ensure a large effective step-size and thus
the R-APA converges fast but results in a large misadjustment error. On the other hand, a large regularization parameter will yield a small effective step-size and thus the R-APA results in small misadjustment error in the steady state, but converges slowly. Along this line of thought we may expect performance improvement through dynamically adjusting the regularization parameter.

### B. Adaptive Regularization Matrix

To achieve this goal, we incorporate a nonidentity regularization matrix which is also dynamically updated so that \( J(\hat{i}) = (1/2)\epsilon^2(\hat{i}) \) is minimized where \( \epsilon(\hat{i}) = \hat{d}(\hat{i}) - \mathbf{w}_i \mathbf{u}_{i-1} \). We start with an APA formulation with a nonidentity regularization matrix \( \mathbf{\Delta}_i \)

\[
\mathbf{w}_i = \mathbf{w}_{i-1} + \mu \mathbf{U}_i^\ast (\mathbf{U}_i \mathbf{U}_i^\ast + \mathbf{\Delta}_i)^{-1} \mathbf{e}_i
\]

where \( \mathbf{\Delta}_i \) is a \( K \times K \) diagonal regularization matrix defined by

\[
\mathbf{\Delta}_i = \text{diag} [\delta_0(\hat{i}), \delta_1(\hat{i}), \ldots, \delta_{K-1}(\hat{i})].
\]

To adapt the regularization parameter, we use a stochastic gradient descent approach which can recursively minimize \( J(\hat{i}) \), i.e.,

\[
\delta_j(\hat{i}) = \delta_j(\hat{i} - 1) - \rho \nabla J(\hat{i})
\]

for \( j = 0, 1, \ldots, K - 1 \) (6)

where \( \rho \) is a small positive learning rate parameter. The gradient of \( J(\hat{i}) \) with respect to \( \delta_j(\hat{i} - 1) \), \( \nabla J(\hat{i}) \), is given by

\[
\nabla J(\hat{i}) = \frac{\partial J(\hat{i})}{\partial \delta(\hat{i})} \cdot \frac{\partial \delta(\hat{i})}{\partial \mathbf{w}_{i-1}} \cdot \frac{\partial \mathbf{w}_{i-1}}{\partial \delta_j(\hat{i} - 1)}.
\]

Each term of right-hand side (RHS) of (7) is simply derived as

\[
\frac{\partial J(\hat{i})}{\partial \delta(\hat{i})} = \epsilon(\hat{i}), \quad \frac{\partial \delta(\hat{i})}{\partial \mathbf{w}_{i-1}} = -\mathbf{u}_i
\]

and

\[
\frac{\partial \mathbf{w}_{i-1}}{\partial \delta_j(\hat{i} - 1)} = -\mu \mathbf{U}_{i-1}^\ast (\mathbf{U}_{i-1} \mathbf{U}_{i-1}^\ast + \mathbf{\Delta}_{i-1})^{-1}
\]

\[
\cdot \frac{\partial \mathbf{U}_{i-1} \mathbf{U}_{i-1}^\ast}{\partial \delta_j(\hat{i} - 1)} (\mathbf{U}_{i-1} \mathbf{U}_{i-1}^\ast + \mathbf{\Delta}_{i-1})^{-1} \mathbf{e}_{i-1}.
\]

More detailed derivation of (8) is given in Appendix. Then we have

\[
\nabla J(\hat{i}) = \mu \epsilon(\hat{i}) \mathbf{u}_i \mathbf{u}_{i-1}^\ast \Gamma_j \mathbf{e}_{i-1}
\]

where we are defining

\[
\Gamma_j = (\mathbf{U}_{i-1} \mathbf{U}_{i-1}^\ast + \mathbf{\Delta}_{i-1})^{-1}
\]

\[
\times \frac{\partial \mathbf{\Delta}_{i-1}}{\partial \delta_j(\hat{i} - 1)} (\mathbf{U}_{i-1} \mathbf{U}_{i-1}^\ast + \mathbf{\Delta}_{i-1})^{-1} \mathbf{e}_{i-1}.
\]

From (6) and (10), note that \( \Delta \delta_j(\hat{i}) = \delta_j(\hat{i}) - \delta_j(\hat{i} - 1) \) is a function of \( \epsilon(\hat{i}) \) and \( |\Delta \delta_j(\hat{i})| \) is proportional to \( |\epsilon(\hat{i})| \) since

\[
|\Delta \delta_j(\hat{i})| = \rho \epsilon(\hat{i}) \cdot |\mathbf{u}_i \mathbf{u}_{i-1}^\ast \Gamma_j \mathbf{e}_{i-1}|.
\]

This implies that a small \( \epsilon(\hat{i}) \) after the initial convergence results in too small change in \( \Delta \delta_j(\hat{i}) \), and correspondingly \( \delta_j(\hat{i}) \) undergoes small variation. This is undesirable since the regularization parameter should increase along with iterations to guarantee the lower misadjustment error.

Motivated by this fact, we normalize the gradient, \( \nabla J(\hat{i}) \), by its norm. Since introducing the normalized gradient always ensures the uniform \( \Delta \delta_j(\hat{i}) \) irrespective of \( |\epsilon(\hat{i})| \), it makes the regularization parameter \( \delta_j(\hat{i}) \) robust to variation of \( \epsilon(\hat{i}) \). This property makes the regularization parameter \( \delta_j(\hat{i}) \) relatively stable when \( \nabla J(\hat{i}) \) is very small. Thus, the regularization parameter \( \delta_j(\hat{i}) \) is recursively updated by

\[
\delta_j(\hat{i}) = \delta_j(\hat{i} - 1) - \rho \frac{\nabla J(\hat{i})}{|\nabla J(\hat{i})|}.
\]

Then, \( \nabla J(\hat{i}) / |\nabla J(\hat{i})| \) in (11) can be rewritten as

\[
\nabla J(\hat{i}) / |\nabla J(\hat{i})| = \text{sgn} (\nabla J(\hat{i}))
\]

where sgn(\cdot) is the signum function which takes the sign of a variable.

From (10)–(12), the proposed AR-APA with the adaptive regularization matrix is summarized as

\[
\delta_j(\hat{i}) = \delta_j(\hat{i} - 1) - \rho \text{sgn} (\mu \epsilon(\hat{i}) \mathbf{u}_i \mathbf{u}_{i-1}^\ast \Gamma_j \mathbf{e}_{i-1})
\]

\[
\delta_j(\hat{i}) = \begin{cases} \delta_j'(\hat{i}), & \text{if } \delta_j'(\hat{i}) \geq \delta_{\text{min}} \\ \delta^\ast_{\text{min}}, & \text{if } \delta_j'(\hat{i}) < \delta_{\text{min}} \end{cases}
\]

\[
\mathbf{w}_i = \mathbf{w}_{i-1} + \mu \mathbf{U}_i^\ast (\mathbf{U}_i \mathbf{U}_i^\ast + \mathbf{\Delta}_i)^{-1} \mathbf{e}_i
\]

where \( \delta_{\text{min}} \) is a minimum allowable value of \( \delta_j(\hat{i}) \).

By setting

\[
\delta(\hat{i}) = \delta_0(\hat{i}) = \cdots = \delta_{K-1}(\hat{i})
\]

we can get a simpler version of the algorithm given in (13d)

\[
\delta'(\hat{i}) = \delta'(\hat{i} - 1) - \rho \text{sgn} \left[ \mu \epsilon(\hat{i}) \mathbf{u}_i \mathbf{u}_{i-1}^\ast \Gamma_j \mathbf{e}_{i-1} \right]
\]

\[
\delta(\hat{i}) = \begin{cases} \delta'(\hat{i}), & \text{if } \delta'(\hat{i}) \geq \delta_{\text{min}} \\ \delta^\ast_{\text{min}}, & \text{if } \delta'(\hat{i}) < \delta_{\text{min}} \end{cases}
\]

\[
\mathbf{w}_i = \mathbf{w}_{i-1} + \mu \mathbf{U}_i^\ast (\mathbf{U}_i \mathbf{U}_i^\ast + \delta(\hat{i}) \mathbf{I})^{-1} \mathbf{e}_i
\]

in which a weighted identity matrix is used for regularization like the conventional R-APA but the weighted identity matrix gets adapted here.

In addition, when \( K = 1 \), we get a adaptive regularized NLMS (AR-NLMS) algorithm [10]. From (13a) or (15a), the AR-NLMS with the adaptive regularization parameter reduces to

\[
\delta'(\hat{i}) = \delta'(\hat{i} - 1) - \rho \text{sgn} \left( \frac{\mu \epsilon(\hat{i}) \epsilon(\hat{i} - 1) \mathbf{u}_i \mathbf{u}_{i-1}^\ast}{||\mathbf{u}_i||^2 + \delta(\hat{i}) - 1)} \right)^2
\]

\[
= \delta(\hat{i} - 1) - \rho \text{sgn} \left( \frac{\epsilon(\hat{i}) \epsilon(\hat{i} - 1) \mathbf{u}_i \mathbf{u}_{i-1}^\ast}{||\mathbf{u}_i||^2 + \delta(\hat{i}) - 1)} \right)
\]

\[
\delta(\hat{i}) = \begin{cases} \delta'(\hat{i}), & \text{if } \delta'(\hat{i}) \geq \delta_{\text{min}} \\ \delta^\ast_{\text{min}}, & \text{if } \delta'(\hat{i}) < \delta_{\text{min}} \end{cases}
\]

\[
\mathbf{w}_i = \mathbf{w}_{i-1} + \mu \mathbf{U}_i^\ast (\mathbf{U}_i \mathbf{U}_i^\ast + \delta(\hat{i}) \mathbf{I})^{-1} \mathbf{e}_i
\]

Table I lists the number of multiplications, additions to compute the adaptive regularization matrix at each iteration. It is known [2] that the complexity of the conventional R-APA is
(\(K^2 + 2K\))M + K^3 + K^2 multiplications and (\(K^2 + 2K\))M + K^3 additions. As can be seen, the additional complexity is relatively little compared to the conventional R-APA.

C. Stability

To guarantee the stability of the proposed AR-APA, we need to set \(\delta_{\text{min}}\). Let us define the \textit{a posteriori} estimation error as

\[
r_i = d_i - U_i w_i
\]

i.e., the error in estimating \(d_i\) by using the new system estimate. Assuming a scalar regularization parameter as (14), it holds that

\[
r_i = \left( I - \mu U_i U_i^\dagger (U_i U_i^\dagger + \delta(i)I)^{-1} \right) e_i
\]

and

\[
|e_i|^2 - |r_i|^2 = e_i^\dagger (I - A^* A) e_i
\]

where we are defining

\[
A = \left( I - \mu U_i U_i^\dagger (U_i U_i^\dagger + \delta(i)I)^{-1} \right) V_i^*.
\]

From (19), the desirable property \(E[|r_i|^2] \leq E[|e_i|^2]\) (with equality only when \(e_i = 0\)) is guaranteed, if and only if the matrix \((I - A^* A)\) is positive-definite. In addition, let \(U_i U_i^\dagger = V_i \Lambda_i V_i^\dagger\) denotes the eigen-decomposition of the matrix \(U_i U_i^\dagger\), where \(V_i\) is unitary and \(\Lambda_i = \text{diag}(\lambda_0(i), \lambda_1(i), \ldots, \lambda_{K-1}(i))\) contains the corresponding eigenvalues. Then

\[
(U_i U_i^\dagger + \delta(i)I)^{-1} = V_i (\Lambda_i + \delta(i)I)^{-1} V_i^*.
\]

Using the eigen-decomposition of \(U_i U_i^\dagger\) and (20), the following holds:

\[
A^* A = (I - \mu V_i \Lambda_i V_i^\dagger) (I - \mu V_i \Lambda_i V_i^\dagger) = I - 2\mu V_i \Lambda_i V_i^\dagger + \mu^2 V_i \Lambda_i^2 V_i^\dagger,
\]

where \(\Lambda_i^\dagger = \Lambda_i (\Lambda_i + \delta(i)I)^{-1}\). So, it holds that

\[
I - A^* A = \mu V_i \Lambda_i (2I - \mu \Lambda_i^\dagger) V_i^\dagger.
\]

To satisfy that \((I - A^* A)\) is positive-definite, \((2I - \mu \Lambda_i^\dagger)\) should be positive. Taking an expectation, we find

\[
E[2I - \mu \Lambda_i^\dagger] = 2I - \mu E \left[ \Lambda_i (\Lambda_i + \delta(i)I)^{-1} \right]
\]

\[
= 2I - \mu E \left[ \frac{1}{\lambda_0(i)} + \frac{1}{\lambda_1(i)} + \ldots + \frac{1}{\lambda_{K-1}(i)} \right] > 0.
\]

Then, we get the lower bound of the regularization parameter for the stability of the proposed AR-APA as

\[
\delta_{\text{min}} > \max_{0 \leq k \leq K-1} E[\lambda_k(i)] \left( \frac{\mu}{2} - 1 \right).
\]

Table I

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>multiplications</th>
<th>additions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed (13a)</td>
<td>(K M + K^2 + 3K + 1)</td>
<td>(K M + K^2 - K)</td>
</tr>
<tr>
<td>Proposed (15a)</td>
<td>(K M + K^2 + 2K + 2)</td>
<td>(K M + K^2 - K)</td>
</tr>
</tbody>
</table>

**III. EXPERIMENTAL RESULTS**

We illustrate the performance of the proposed AR-APA by carrying out computer simulations in a channel identification in which the unknown channel is randomly generated. The length of the unknown channel is \(M = 16\) in experiments except for Figs. 4 and 8 where \(M = 128\) is used. We compare the adaptive scalar regularization parameter in (15c) as expected. Two Gaussian distributed signals are used for the input signal. The input signals are obtained by filtering a white, zero-mean, Gaussian random sequence \(\alpha(i)\) through a first-order system \(G_1(z) = 1/(1 - 0.9z^{-1})\) or a second-order system \(G_2(z) = 1 + 0.5z^{-1} + 0.81z^{-2}\). The signal-to-noise ratio (SNR) is calculated by

\[
\text{SNR} = 10 \log_{10} \left( \frac{E[y^2(i)]}{E[n^2(i)]} \right)
\]

where \(y(i) = u_i w^\dagger\). The measurement noise \(v(i)\) is added to \(y(i)\) such that \(\text{SNR} = 30\) dB. The mean square deviation (MSD), \(E[|w^\dagger - w_i|^2]\), is taken and averaged over 100 independent trials. The initial value \(\delta(0)\) is set to 0.001 and \(\delta_{\text{min}}\) is chosen to 0.0001 for all experiments. The step-size is set to \(\mu = 1.0\) and the learning rate parameter \(\rho\) is set to \(\rho = 1.0\), respectively. We used the input signals generated by \(G_1(z)\) and \(G_2(z)\) for Figs. 1–5 and Figs. 6–9, respectively.

In Fig. 1, it shows the MSD curves for \(K = 8\). Dashed lines indicate the results of the R-APA with fixed regularization parameters where we choose \(\delta = 0.001\) and 300. As can be seen, the proposed AR-APA has the faster convergence and the lower misadjustment error than the conventional R-APA. In addition, the proposed AR-APA with the adaptive nonidentity regularization matrix in (13d) has improved performance over with the adaptive scalar regularization parameter in (15c) as expected. Fig. 2 shows the performance of the proposed AR-NLMS and

**TABLE I**

COMPUTATIONAL COMPLEXITY OF ADAPTIVE REGULARIZATION MATRIX

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>multiplications</th>
<th>additions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed (13a)</td>
<td>(K M + K^2 + 3K + 1)</td>
<td>(K M + K^2 - K)</td>
</tr>
<tr>
<td>Proposed (15a)</td>
<td>(K M + K^2 + 2K + 2)</td>
<td>(K M + K^2 - K)</td>
</tr>
</tbody>
</table>
This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Fig. 2. MSD curves of the proposed AR-NLMS in (16c), GNGD [9], and conventional R-NLMS [J = 1, M = 16, Input: Gaussian AR(1), pole at 0.9].

Fig. 3. MSD curves of the proposed AR-APA in (13d), R-APA with delta-control [8], and RLS [K = 8, M = 16, Input: Gaussian AR(1), pole at 0.9].

Fig. 4. MSD curves of the proposed AR-APA in (13d), R-APA with delta-control [8], and conventional R-APA [K = 4, M = 128, Input: Gaussian AR(1), pole at 0.9].

Fig. 5. MSD curves of the proposed AR-APA in (13d) and conventional R-APA. The channel is suddenly changed from w to -w [K = 8, M = 16, Input: Gaussian AR(1), pole at 0.9].

Fig. 6. MSD curves of the proposed AR-APA in (13d) and (15c), and conventional R-APA [K = 8, M = 16, Input: Gaussian ARMA(2,2)].

Fig. 7. MSD curves of the proposed AR-NLMS in (16c), GNGD [9], and conventional R-NLMS [K = 1, M = 16, Input: Gaussian ARMA(2,2)].

the conventional R-NLMS, i.e., K = 1. For comparison purposes, the generalized normalized gradient descent (GNGD) algorithm [9] is presented using same ρ. A similar result of Fig. 1 is observed in Fig. 2. Fig. 3 demonstrates the performance comparison of the proposed AR-APA, the R-APA with the delta-control method [8] where K = 8, and the RLS. We choose the forgetting factor as λ = 0.8 and 0.995 for the RLS. In the figure, we can see that the proposed AR-APA outperforms the R-APA with the delta-control method. In addition, the AR-APA is comparable to the RLS in that the AR-APA has
the similar convergence rate and misadjustment with smaller \( \lambda = 0.8 \) and larger \( \lambda = 0.995 \), respectively. In Fig. 4, it shows the MSD curves for the proposed AR-APA compared to other APAs when higher filter length is used (\( M = 128 \)). The order of the APA is set to \( K = 4 \). We can see that a similar result of \( M = 16 \) is observed in Fig. 4. Now, we examine the tracking capability of the AR-APA to a sudden change in the channel. Fig. 5 shows the result of suddenly multiplying the unknown channel by \( -1 \). It can be seen that the AR-APA keeps track of weight change without losing the convergence speed nor the misadjustment.

Figs. 6–9 are the experimental results with the different input signal generated by \( G_2(z) \).

IV. CONCLUSION

We have proposed an APA with adaptive regularization matrix which improves performance in both the convergence speed and the misadjustment error. Although several scalar regularization methods have been exploited for the APA, to the best of authors’ knowledge, a matrix regularization in the APA has not been discussed yet. Also for performance optimization, we dynamically updated the regularization matrix. As a result, the proposed AR-APA has achieved highly improved performance.

APPENDIX

DERIVATION OF (9)

We start with

\[
\frac{\partial w_{i-1}}{\partial \delta_j(i-1)} = \mu \frac{\partial U_{i-1}^a (U_{i-1} U_{i-1}^a + \Delta_{i-1})^{-1} e_{i-1}}{\partial \delta_j(i-1)}
\]

\[
= \mu U_{i-1}^a \frac{\partial (U_{i-1} U_{i-1}^a + \Delta_{i-1})^{-1} e_{i-1}}{\partial \delta_j(i-1)}.
\]  

(26)

Let us define

\[
Y \triangleq (U_{i-1} U_{i-1}^a + \Delta_{i-1}).
\]

From [11], we know that by differentiating \( YY^{-1} = I \) with respect to \( \delta_j(i-1) \)

\[
\frac{\partial Y}{\partial \delta_j(i-1)} Y^{-1} + Y^{-1} \frac{\partial Y^{-1}}{\partial \delta_j(i-1)} = 0,
\]

(27)

Then, we get

\[
\frac{\partial Y^{-1}}{\partial \delta_j(i-1)} = -Y^{-1} \frac{\partial Y}{\partial \delta_j(i-1)} Y^{-1}.
\]  

(28)

Now we substitute \( Y = (U_{i-1} U_{i-1}^a + \Delta_{i-1}) \) into (28), then

\[
\frac{\partial (U_{i-1} U_{i-1}^a + \Delta_{i-1})^{-1}}{\partial \delta_j(i-1)}
\]

\[
= - (U_{i-1} U_{i-1}^a + \Delta_{i-1})^{-1} \frac{\partial (U_{i-1} U_{i-1}^a + \Delta_{i-1})}{\partial \delta_j(i-1)}
\]

\[
\cdot (U_{i-1} U_{i-1}^a + \Delta_{i-1})^{-1}
\]

\[
= - (U_{i-1} U_{i-1}^a + \Delta_{i-1})^{-1} \frac{\partial \Delta_{i-1}}{\partial \delta_j(i-1)}
\]

\[
\cdot (U_{i-1} U_{i-1}^a + \Delta_{i-1})^{-1}.
\]  

(29)

Using (26) and (29), we obtain (9).

REFERENCES


