Systolic arrays for matrix I/O format conversion
Petkov, Nicolai

Published in:
Default journal

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
1988

Link to publication in University of Groningen/UMCG research database

Citation for published version (APA):

Copyright
Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

Take-down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): http://www.rug.nl/research/portal. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.
SYSTOLIC ARRAYS FOR MATRIX I/O FORMAT CONVERSION

Indexing terms: Computer applications, Systolic arrays, Matrix algebra, I/O format conversion

New 1-slow and 2-slow systolic arrays for row-to-diagonal and diagonal-to-row input/output format conversion of matrices are given. For \( n \times n \) matrices, the arrays consist of \( n^2 \) cells. A 1-slow systolic array performs the conversion in \( 2n - 1 \) periods. The arrays can be used also for column-to-diagonal, diagonal-to-column, row-to-column and column-to-row input/output format conversions. The 1-slow array performs these operations in \( 2n - 1 \), \( 2n - 1 \), and \( 3n - 1 \) periods, respectively.

Systolic arrays for several important matrix operations have been a matter of wide interest in the last few years.\(^1\) The matrices involved are input/output in different formats. We refer to an input format in which the elements of one matrix row are input one after another in the same array input as row input format. (Different matrix rows can be input in different array inputs.) Column and diagonal input/output formats are defined by analogy. An orthogonal systolic array\(^2\) for the matrix operation \( C = A \cdot B + C^0 \) is a representative example in which all three (row, column, and diagonal) I/O formats are used. Systolic algorithms for I/O format conversion have been proposed by O'Leary.\(^3\) In this letter, new systolic arrays for row-to-diagonal and diagonal-to-row I/O format conversion are presented. They can also be used for column-to-diagonal, diagonal-to-column, row-to-column, and column-to-row I/O format conversion.

The arrays given make use of a switch cell shown in Fig. 1.

It is a combinational circuit taking in matrix components \( x \) and \( y \) from the left and from below, respectively, and a control bit \( c \) from the right. If \( c \) is zero, then \( x \) is output to the right and \( y \) up. If \( c = 1 \), then \( x \) is output up and \( y \) to the right.

A systolic array for row-to-diagonal format conversion of \( 3 \times 3 \) matrices is shown in Fig. 2. The small black boxes denote clocked delay elements. In this particular algorithm, the zero control bits which follow the one control bits propagating from right to left have no influence on the computational processes, but they are provided to prepare the array for a subsequent computation. (Notice that at the beginning of the computation, a zero control bit should reside in the second cell of the last array row and in the last cell of the second row, in order that the array functions properly. These zero bits can be considered as 'a nonobligatory rest' from a previous computation.) The reader is invited to check the correctness of this and the other algorithms given below by simulating the array's work for several clock periods. For \( n \times n \) matrices, the conversion requires \( n^2 \) cells and \( 3n - 2 \) clock periods. The array given in Fig. 2 is classified as a 2-slow circuit (after the terminology of Leiserson et al.),\(^4\) and can readily be transformed into a 1-slow circuit by removing every second delay element in each directed interconnection path (Fig. 3). The data input should be changed, the most significant change being the removal of the idle periods between consecutive matrix elements. The 1-slow systolic array for row-to-diagonal conversion of an \( n \times n \) matrix consists of \( n^2 \) cells and performs the conversion in \( 2n - 1 \) clock periods.

Fig. 3 1-slow systolic array for row-to-diagonal format conversion (\( n = 3 \)).

Fig. 2 2-slow systolic array for row-to-diagonal format conversion (\( n = 3 \)).
lapped with the first $n$ periods of the second operation, so that the row-to-column conversion can be performed in $3n - 2$ clock periods.

Next, the performance of the (1-slow) algorithms given above is compared with alternative designs. O'Leary’s has given three different systolic arrays for row-to-column, row-to-diagonal, and diagonal-to-row format conversion, respectively. They perform the conversion in $3n - 1$, $4n - 1$, and $2n - 1$ clock periods, respectively, while the 1-slow systolic array given above requires $3n - 2$, $2n - 1$, and $2n - 1$ clock periods, respectively. The 1-slow systolic array given above has the following advantages: one and the same array is used for all operations at internal cells and the row-to-diagonal conversion is twice as fast.

International Basic Laboratory
Central Institute for Cybernetics
Kurstr. 33, PO Box 1298, Berlin 1086, German Democratic Republic

References


PERFORMANCE ANALYSIS OF CA-CFAR IN THE PRESENCE OF COMPOUND GAUSSIAN CLUTTER

Indexing terms: Radar, Clutter, CFAR, Radar receivers

Exact expressions are derived for the ROCs of CA-CFAR radar processor subject to both internal noise and clutter (last one modelled as compound Gaussian). They include previously known ROCs, e.g. for ideal CFAR. The effect of finite sample size is elicited in the example of K-distributed clutter.

Non-Gaussian clutter is of primary concern for high-resolution radars. Deviations from the Gaussian statistics may be considered to arise from the concurrence of two reflectivity phenomena: the first one is a slow fluctuation of backscatter coefficient induced by physical variations of the scanned domain; the second one is a superimposed fast fluctuation for fixed backscatter coefficient. If the resolution cell is small enough the radar return is conditionally Gaussian (conditioned on a fixed illuminated patch), but its overall distribution may be non-Gaussian since it is affected by the statistics of the slow fluctuation. This is the so-called composite scattering model.

Correspondingly a mathematical model for the complex envelope $\hat{x}(t)$ of the clutter is the so-called compound Gaussian model:

$$\hat{x}(t) = \alpha x(t)$$  (1)

where $\hat{x}(t)$, the ‘speckle’ component, is a Gaussian complex process with zero mean and variance $\sigma^2$ accounting for the fast fluctuation; $x(t)$, the ‘spiky’ component, is a random variable, independent of $\hat{x}(t)$, whose distribution fits the overall statistics of the slow fluctuation. In considering $x(t)$ as a random variable instead of a random process the assumption is made that the illuminated patch is small enough that the time slow fluctuation can be neglected. To achieve constant false alarm rate in such a time-varying and nonhomogeneous environment some sort of adaptivity is required. We focus on the adaptive thresholding technique known as cell averaging (CA). The CA-CFAR basic scheme is shown in Fig. 1 with reference to averaging in range. Following square-law detection, echoes from a number of cells surrounding and including the cell being probed are used to estimate the background noise, so that the detection threshold can be adapted accordingly.

Assuming that the useful signal $s(t)$ fluctuates from pulse to pulse according to a Swerling model and considering receiver noise $n(t)$ in addition to clutter, the hypothesis testing problem can be formulated as follows:

$$H_0: \hat{r}(t) = \hat{n}(t) + \alpha \hat{x}(t)$$  (2)

$$H_1: \hat{r}(t) = \hat{n}(t) + \alpha \hat{x}(t) + \hat{x}(t)$$  (3)

where $\hat{n}(t)$, $\hat{x}(t)$, and $\hat{x}(t)$ are complex zero-mean Gaussian variables with variances $\sigma^2_x$, $\sigma^2$ and $\sigma^2$ respectively and $\alpha$ has unit RMS.

The performance of this basic scheme has been previously considered in References 2 and 3 for the case that the clutter envelope is $K$-distributed. The assumption is made there that the speckle component is completely averaged out by the CA process or, equivalently, that the spiky component is completely known: this is the 'ideal CFAR' since it implies constancy of the clutter (constant $\alpha$) in an arbitrary large sample set. Following such an approach, the operating characteristics

Fig. 4 2-slow systolic array for diagonal-to-row format conversion ($n = 3$)

Fig. 1 Schema of cell averaging constant false alarm rate processor