Mathematical Geology, Vol. 34, No. 5, July 2002 (© 2002)

Conditional Simulation of Random Fields by Successive Residuals[1]

J. A. Vargas-Guzmán[2,3] and R. Dimitrakopoulos[2]

This paper presents a new approach to the **LU** *decomposition method for the simulation of stationary and ergodic random fields. The approach overcomes the size limitations of* **LU** *and is suitable for any size simulation. The proposed approach can facilitate fast updating of generated realizations with new data, when appropriate, without repeating the full simulation process. Based on a novel column partitioning of the* **L** *matrix, expressed in terms of successive conditional covariance matrices, the approach presented here demonstrates that* **LU** *simulation is equivalent to the "successive" solution of kriging residual estimates plus random terms. Consequently, it can be used for the* **LU** *decomposition of matrices of any size. The simulation approach is termed "conditional simulation by successive residuals" as at each step, a small set (group) of random variables is simulated with a* **LU** *decomposition of a matrix of updated conditional covariance of residuals. The simulated group is then used to estimate residuals without the need to solve large systems of equations.*

KEY WORDS: conditional simulation, **LU** decomposition, successive conditional covariances.

INTRODUCTION

Modelling of spatial data in earth sciences and engineering is often based on the conditional simulation of stationary and ergodic Gaussian random fields. A well known Gaussian conditional simulation based on the lower–upper (**LU**) decomposition of the covariance matrix **C** of data and grid node locations was introduced into geostatistics (Davis, 1987a). Conditional simulation by **LU** decomposition may be an attractive method because of its efficiency, simplicity, and simultaneous conditioning to available data during the simulation. More specifically, let a set $\mathbf{\Omega}_d$ of sample locations correspond to a data vector $\mathbf{z}_d$. A realization $\mathbf{z}$ of a spatial random field $Z(x)$, $x \in R^n$, at a set $\mathbf{\Omega}_g$ of q grid node locations conditional to $\mathbf{\Omega}_d$, is a vector generated from

$$\mathbf{z} = \mathbf{Lw} \tag{1}$$

[1]Received 3 November 2000; accepted 24 January 2002.

[2]WH Bryan Mining Geology Research Centre, The University of Queensland, Brisbane, Queensland 4072, Australia; e-mail: brc@mailbox.uq.edu.au

[3]e-mail: a.varguz@uq.edu.au

where **w** is a vector of white noise and **L** is

$$\mathbf{L} = \mathbf{C}\mathbf{U}^{-1} \tag{2}$$

The size of matrix **L** poses the well-known limitation to the application of the method to the generation of realizations up to only few thousand grid nodes.

Using the matrix form of kriging to simultaneously estimate set Ω_g with the entire sample set Ω_d, it has been shown (Alabert, 1987; Davis, 1987a) that **LU** can be written as the sum of a simple estimate plus a random component,

$$\mathbf{z} = \mathbf{A}_{21}\mathbf{L}_{11}^{-1}\mathbf{z}_d + \mathbf{L}_{22}\mathbf{w} \tag{3}$$

where the partitioning $\mathbf{L} = \begin{bmatrix} \mathbf{L}_{11} & 0 \\ \mathbf{A}_{21} & \mathbf{L}_{22} \end{bmatrix}$ is considered. The $\mathbf{L}_{11}$ matrix is derived from the **LU** decomposition of the data covariance matrix. The $\mathbf{A}_{21}$ and $\mathbf{L}_{22}$ matrices are obtained from the partitioned **L** matrix, generated by the simultaneous decomposition of the covariance **C** in Equation (2). Based on the decomposition in Equation (2), Equation (3) also has the same computational limitations in generating realizations on few thousand nodes at a time.

To enhance **LU**, Davis (1987b) suggests replacing the **L** matrix in Equation (1) by the square root **B** of the covariance **C**. If **C** is symmetric and positive definite, matrix **B** can be computed by a minimax polynomial numerical approximation. This approach does not allow for simultaneous simulation and conditioning, and it does not guarantee computational stability. Alabert (1987) proposes the use of overlapping windows in **LU**, a solution that generates discontinuities in realizations. Dietrich and Newsam (1995) extend the approach by Davis (1987b) using Chebyshev matrix polynomial approximations of the square root matrices **B**. Side effects of this approach are a decrease in computational speed and conditioning can only accommodate a small data set. Dowd and Saraç (1993) improve **LU** using a ring decomposition that extends the upper limit for **LU** to a several thousand points, without resolving the general issue of matrix size.

A related development is sequential Gaussian simulation (SGS) (Isaaks, 1990; Journel, 1994). SGS is based on the decomposition of the multivariate probability density function of a Gaussian random field (Johnson, 1987) and does not have the size limitations of **LU** decomposition. The method conceptually follows the explanation in Equation (3) that each simulated value is the sum of a kriged value plus a simulated spatially correlated error. The algorithm performs kriging at a randomly chosen node to estimate the conditional mean and variance. A random residual is then drawn from a conditional distribution and is added to the conditional mean to provide a realization at the corresponding node. This simulated

value is appended to the data set when moving from one node to the next. The implementation of the technique may require multiple size grids (Deutsch and Journel, 1998; Isaaks, 1990) that are used to simulate first a coarse grid and subsequently finer grids to ensure the reproduction of the global covariance. An additional characteristic of available simulation methods is that updating of existing realizations with new additional data is only available by repeating the whole simulation process. A new possible size unlimited **LU** approach could facilitate a simpler and more efficient conditional simulation process and updating of existing realizations.

This paper presents a novel alternative and new formulation of the **LU** decomposition simulation method termed conditional simulation by successive residuals (CSSR). The method does not have the size limits of traditional simulation by **LU** and, at the same time, it can accommodate a relatively simple updating of a simulated realization if additional data becomes available. The approach is developed from the partitioning of the **L** matrix in **LU** by columns using conditional covariance matrices. The column partitioning of the **L** matrix leads to a theoretical link between simulation via **LU** decomposition and the successive estimation of residuals plus a generated random error. The residual estimates are found to be equivalent to kriging estimates derived from a successive minimization of the estimation variance (Vargas-Guzmán and Yeh, 1999). The column partitioning of the **L** matrix leads to the development of the simulation approach suitable for generating large realizations without having to solve large kriging systems, while using all of the available data for conditioning. In addition, the residual estimates in the solution generated for the **LU** decomposition allow for the fast updating of old realizations when more data become available.

In the following sections the partition of **L** is developed and linked to successive conditional covariances. This leads to a new form of the **L** matrix based on conditional covariances. The extension to the multivariate case follows. Finally, the resulting conditional simulation algorithm CSSR based on the successive simulation of residuals is presented.

PROPOSED METHOD

A Partitioned View of LU

Consider generating a large realization of a stationary and ergodic random function $Z(x)$. The conditioning data is from the set of locations $\boldsymbol{\Omega}_d$ and is split into subsets as $\boldsymbol{\Omega}_d = \boldsymbol{\Omega}_p \cup \boldsymbol{\Omega}_s \cup \cdots \cup \boldsymbol{\Omega}_t$. The first or prior subset of data is $\boldsymbol{\Omega}_p$, a second subset of data is $\boldsymbol{\Omega}_s$ and so forth. Since the final conditioning should be independent of the order in which the samples are utilized, there is no definitive numerical sequence in this split. The corresponding partitioned sample covariance

matrix is

$$\mathbf{C}_d = \begin{bmatrix} \mathbf{C}_{pp} & \mathbf{C}_{ps} & \mathbf{C}_{pr} & \mathbf{C}_{pu} & \cdots & \mathbf{C}_{pt} \\ \mathbf{C}_{sp} & \mathbf{C}_{ss} & \mathbf{C}_{sr} & \mathbf{C}_{su} & \ldots & \mathbf{C}_{st} \\ \mathbf{C}_{rp} & \mathbf{C}_{rs} & \mathbf{C}_{rr} & \mathbf{C}_{ru} & \cdots & \mathbf{C}_{rt} \\ \mathbf{C}_{up} & \mathbf{C}_{us} & \mathbf{C}_{ur} & \mathbf{C}_{uu} & \cdots & \mathbf{C}_{ut} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ \mathbf{C}_{tp} & \mathbf{C}_{ts} & \mathbf{C}_{tr} & \mathbf{C}_{tu} & \cdots & \mathbf{C}_{tt} \end{bmatrix} \tag{4}$$

The set of grid nodes $\boldsymbol{\Omega}_g$ simulated is split into m subsets $\boldsymbol{\Omega}_g = \boldsymbol{\Omega}_v \cup \cdots \cup \boldsymbol{\Omega}_m$ each one made of random locations. The subscripts $v, \ldots, m$ identify each subset for simulation and there is no numerical sequence to be followed as the order changes from one realization to another. Instead of thinking of one point at a time, a set of locations may be spread out along the simulated domain. It is also possible that a group of simulated points may be within a block or a spatial cluster. The partitioned covariance matrix made of the matrices of the simulated m sets of nodes is given as

$$\mathbf{C}_g = \begin{bmatrix} \mathbf{C}_{vv} & \cdots & \mathbf{C}_{vm} \\ \vdots & & \vdots \\ \mathbf{C}_{mv} & \cdots & \mathbf{C}_{mm} \end{bmatrix} \tag{5}$$

The global set, including sample data locations and simulated grid nodes, is $\boldsymbol{\Omega}_G = \boldsymbol{\Omega}_d \cup \boldsymbol{\Omega}_g$, and a covariance matrix $\mathbf{C}_G$ is constructed. This is,

$$\mathbf{C}_G = \mathbf{LU} = \begin{bmatrix} \begin{bmatrix} \mathbf{C}_{pp} & \mathbf{C}_{ps} & \mathbf{C}_{pr} & \mathbf{C}_{pu} & \cdots & \mathbf{C}_{pt} \\ \mathbf{C}_{sp} & \mathbf{C}_{ss} & \mathbf{C}_{sr} & \mathbf{C}_{su} & \cdots & \mathbf{C}_{st} \\ \mathbf{C}_{rp} & \mathbf{C}_{rs} & \mathbf{C}_{rr} & \mathbf{C}_{ru} & \cdots & \mathbf{C}_{rt} \\ \mathbf{C}_{up} & \mathbf{C}_{us} & \mathbf{C}_{ur} & \mathbf{C}_{uu} & \cdots & \mathbf{C}_{ut} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ \mathbf{C}_{tp} & \mathbf{C}_{ts} & \mathbf{C}_{tr} & \mathbf{C}_{tu} & \cdots & \mathbf{C}_{tt} \end{bmatrix} & \begin{bmatrix} \mathbf{C}_{pv} & \cdots & \mathbf{C}_{pm} \\ \mathbf{C}_{sv} & \cdots & \mathbf{C}_{sm} \\ \mathbf{C}_{rv} & \cdots & \mathbf{C}_{rm} \\ \mathbf{C}_{uv} & \cdots & \mathbf{C}_{um} \\ \vdots & & \vdots \\ \mathbf{C}_{tv} & \cdots & \mathbf{C}_{tm} \end{bmatrix} \\ \begin{bmatrix} \mathbf{C}_{vp} & \mathbf{C}_{vs} & \mathbf{C}_{vr} & \boldsymbol{C}_{vu} & \cdots & \mathbf{C}_{vt} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ \mathbf{C}_{mp} & \mathbf{C}_{ms} & \mathbf{C}_{mr} & \mathbf{C}_{mu} & \cdots & \mathbf{C}_{mt} \end{bmatrix} & \begin{bmatrix} \mathbf{C}_{vv} & \cdots & \mathbf{C}_{vm} \\ \vdots & & \vdots \\ \mathbf{C}_{mv} & \cdots & \mathbf{C}_{mm} \end{bmatrix} \end{bmatrix} \tag{6}$$

Equation (6) is a global covariance matrix utilized in simultaneous **LU** decomposition. However, it considers partitioning of the conditioning data and the simulated

set as $\mathbf{\Omega}_G = \mathbf{\Omega}_p \cup \mathbf{\Omega}_s \cup \mathbf{\Omega}_r \cup \mathbf{\Omega}_u \cup \cdots \cup \mathbf{\Omega}_t \cup \mathbf{\Omega}_v \cup \cdots \cup \mathbf{\Omega}_m$. An alternative way of partitioning $\mathbf{C}_G$ is by columns following $\mathbf{\Omega}_G \cap \mathbf{\Omega}_j$ where the subscript are $j = (p, s, r, u, \ldots t, v, \ldots m)$, this is

$$\mathbf{C}_G = \lfloor \mathbf{C}_{Gp}\ \mathbf{C}_{Gs}\ \mathbf{C}_{Gr}\ \mathbf{C}_{Gu} \cdots \mathbf{C}_{Gt}\ \mathbf{C}_{Gv} \cdots \mathbf{C}_{Gm} \rfloor \tag{7}$$

The $\mathbf{L}$ and $\mathbf{U}$ matrices can also be written in a partitioned manner without considering a computation method for each term yet. This is expressed as

$$\mathbf{L} = \begin{bmatrix} \begin{bmatrix} \mathbf{L}_{pp} & & & & \\ \mathbf{A}_{sp} & \mathbf{L}_{ss} & & & \\ \mathbf{A}_{rp} & \mathbf{B}_{rs} & \mathbf{L}_{rr} & & \\ \mathbf{A}_{up} & \mathbf{B}_{us} & \mathbf{D}_{ur} & \mathbf{L}_{uu} & \\ \vdots & \vdots & \vdots & \vdots & \\ \mathbf{A}_{tp} & \mathbf{B}_{ts} & \mathbf{D}_{tr} & \mathbf{E}_{tu} & \cdots\ \mathbf{L}_{tt} \end{bmatrix} & \\ \begin{bmatrix} \mathbf{A}_{vp} & \mathbf{B}_{vs} & \mathbf{D}_{vr} & \mathbf{E}_{vu} & \cdots\ \mathbf{F}_{vt} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{A}_{mp} & \mathbf{B}_{ms} & \mathbf{D}_{mr} & \mathbf{E}_{mu} & \cdots\ \mathbf{F}_{mt} \end{bmatrix} & \begin{bmatrix} \mathbf{L}_{vv} & \\ \vdots & \\ \mathbf{G}_{mv} & \cdots\ \mathbf{L}_{mm} \end{bmatrix} \end{bmatrix} \tag{8}$$

$$\mathbf{U} = \begin{bmatrix} \begin{bmatrix} \mathbf{U}_{pp} & \mathbf{H}_{ps} & \mathbf{H}_{pr} & \mathbf{H}_{pu} & \cdots\ \mathbf{H}_{pt} \\ & \mathbf{U}_{ss} & \mathbf{M}_{sr} & \mathbf{M}_{su} & \cdots\ \mathbf{M}_{st} \\ & & \mathbf{U}_{rr} & \mathbf{N}_{ru} & \cdots\ \mathbf{N}_{rt} \\ & & & \mathbf{U}_{uu} & \cdots\ \mathbf{R}_{ut} \\ & & & & \vdots \\ & & & & \mathbf{U}_{tt} \end{bmatrix} & \begin{bmatrix} \mathbf{H}_{pv} & \cdots\ \mathbf{H}_{pm} \\ \mathbf{M}_{sv} & \cdots\ \mathbf{M}_{sm} \\ \mathbf{N}_{rv} & \cdots\ \mathbf{N}_{rm} \\ \mathbf{R}_{uv} & \cdots\ \mathbf{R}_{um} \\ \vdots & \vdots \\ \mathbf{Q}_{tv} & \cdots\ \mathbf{Q}_{tm} \end{bmatrix} \\ & \begin{bmatrix} \mathbf{U}_{vv} & \cdots\ \mathbf{W}_{vm} \\ & \vdots \\ & \mathbf{U}_{mm} \end{bmatrix} \end{bmatrix} \tag{8}$$

where $\mathbf{L}$ can also be partitioned in columns

$$\mathbf{L} = \lfloor \mathbf{A}_{Gp}\ \mathbf{B}_{Gs}\ \mathbf{D}_{Gr}\ \mathbf{E}_{Gu} \cdots \mathbf{F}_{Gt}\ \mathbf{G}_{Gv} \cdots \mathbf{L}_{Gm} \rfloor \tag{9}$$

For convenience $\mathbf{L}$ is computed by Cholesky decomposition.

Successive Conditional Covariances and LU

An analysis of the covariance matrices in terms of the partitioned **LU** matrices leads to fundamental relationships. From Equations (6) and (8)

$$\mathbf{C}_{pp} = \mathbf{L}_{pp}\mathbf{U}_{pp} \tag{10}$$

and

$$\mathbf{C}_{sp} = \mathbf{A}_{sp}\mathbf{U}_{pp} \tag{11}$$

Combining Equations (10) and (11) and considering $\mathbf{C}_{ps} = \mathbf{C}_{sp}^{\mathrm{T}}$ leads to a first kriging as

$$\mathbf{C}_{sp}\mathbf{C}_{pp}^{-1} = \mathbf{A}_{sp}\mathbf{L}_{pp}^{-1} = \boldsymbol{\Lambda}_{ps}^{1} \tag{12}$$

where $\boldsymbol{\Lambda}_{ps}^{1}$ is a matrix of kriging weights when just the subset of data $\boldsymbol{\Omega}_p$ is utilized to estimate the random variables at $\boldsymbol{\Omega}_s$ locations. Notice that the subset of locations $\boldsymbol{\Omega}_s$ may be substituted by any other subset and following partitioning by columns in Equation (7) this is

$$\mathbf{C}_{Gp}\mathbf{C}_{pp}^{-1} = \mathbf{A}_{Gp}\mathbf{L}_{pp}^{-1} = \boldsymbol{\Lambda}_{pG}^{1} \tag{13}$$

where $\mathbf{A}_{Gp}$ is the first column of matrices in **L** from Equation (9). From Equations (6) and (8) the covariance within the set $\boldsymbol{\Omega}_s$ is

$$\mathbf{C}_{ss} = \mathbf{A}_{sp}\mathbf{H}_{ps} + \mathbf{L}_{ss}\mathbf{U}_{ss} \tag{14}$$

and

$$\mathbf{C}_{ps} = \mathbf{L}_{pp}\mathbf{H}_{ps} \tag{15}$$

Using Equations (12) and (15), Equation (14) yields

$$\mathbf{L}_{ss}\mathbf{U}_{ss} = \mathbf{C}_{ss} - \mathbf{C}_{sp}\mathbf{C}_{pp}^{-1}\mathbf{C}_{ps} = \boldsymbol{\xi}_{ss} \tag{16}$$

where $\boldsymbol{\xi}_{ss}$ is the conditional covariance matrix for residuals within subset $\boldsymbol{\Omega}_s$.

The analysis proceeds with a following covariance. Equations (6) and (8) give

$$\mathbf{C}_{rs} = \mathbf{A}_{rp}\mathbf{H}_{ps} + \mathbf{B}_{rs}\mathbf{U}_{ss} \tag{17}$$

Rearranging this in terms of the above equations yields

$$\mathbf{B}_{rs}\mathbf{U}_{ss} = \mathbf{C}_{rs} - \mathbf{C}_{rp}\mathbf{C}_{pp}^{-1}\mathbf{C}_{ps} = \boldsymbol{\xi}_{rs} \tag{18}$$

Using Equations (15) and (18) yields a second kriging of conditional residuals as

$$\mathbf{B}_{rs}\mathbf{L}_{ss}^{-1} = \boldsymbol{\xi}_{rs}\boldsymbol{\xi}_{ss}^{-1} = \boldsymbol{\Lambda}_{sr}^{2} \tag{19}$$

Notice $\boldsymbol{\Omega}_t$ can be substituted by any other subset and in general for the remaining grid one gets

$$\mathbf{B}_{Gs}\mathbf{L}_{ss}^{-1} = \boldsymbol{\xi}_{Gs}\boldsymbol{\xi}_{ss}^{-1} = \boldsymbol{\Lambda}_{sG}^{2} \tag{20}$$

where $\mathbf{B}_{Gs}$ is the second column of matrices $\mathbf{L}$ in Equation (8) and $\boldsymbol{\Lambda}_{sG}^{2}$ is a matrix of kriging weights for the second column.

The analysis continues by looking at the global partitioned matrices of Equations (6) and (8). From the product of row u in $\mathbf{L}$ and column r in $\mathbf{U}$ of Equation (8), a following term is

$$\mathbf{C}_{ur} = \mathbf{A}_{up}\mathbf{H}_{pr} + \mathbf{B}_{us}\mathbf{M}_{sr} + \mathbf{D}_{ur}\mathbf{U}_{rr} \tag{21}$$

From the above results $\mathbf{A}_{up}\mathbf{H}_{pr} = \mathbf{C}_{up}\mathbf{C}_{pp}^{-1}\mathbf{C}_{pr}$, and $\mathbf{L}_{ss}\mathbf{M}_{sr} = \mathbf{C}_{sr} - \mathbf{A}_{sp}\mathbf{H}_{pr} = \mathbf{C}_{sr} - \mathbf{C}_{sp}\mathbf{C}_{pp}^{-1}\mathbf{C}_{pr}$ and using Equation (20) yields

$$\mathbf{D}_{ur}\mathbf{U}_{rr} = \mathbf{C}_{ur} - \mathbf{C}_{up}\mathbf{C}_{pp}^{-1}\mathbf{C}_{pr} - \boldsymbol{\xi}_{us}\boldsymbol{\xi}_{ss}^{-1}\boldsymbol{\xi}_{sr} \tag{22}$$

This is a new updated conditional covariance such that

$$\mathbf{D}_{ur}\mathbf{U}_{rr} = \boldsymbol{\xi}_{ur} - \boldsymbol{\xi}_{us}\boldsymbol{\xi}_{ss}^{-1}\boldsymbol{\xi}_{sr} \tag{23}$$

which may be generalized for a column j and two locations i and k as a successive residual conditional covariance. This is

$$\boldsymbol{\xi}_{ik}^{j+1} = \boldsymbol{\xi}_{ik}^{j} - \boldsymbol{\xi}_{ij}^{j}\left[\boldsymbol{\xi}_{jj}^{j}\right]^{-1}\boldsymbol{\xi}_{jk}^{j} \tag{24}$$

and at a $j+1$ column t in $\mathbf{L}$ the kriging of residuals is

$$\mathbf{F}_{kt}\mathbf{L}_{tt}^{-1} = \boldsymbol{\xi}_{kt}^{j+1}\left[\boldsymbol{\xi}_{tt}^{j+1}\right]^{-1} = \boldsymbol{\Lambda}_{tk}^{j+1} \tag{25}$$

using Equation (7) this yields

$$\mathbf{F}_{Gt}\mathbf{L}_{tt}^{-1} = \boldsymbol{\xi}_{Gt}^{j+1}\left[\boldsymbol{\xi}_{tt}^{j+1}\right]^{-1} = \boldsymbol{\Lambda}_{tG}^{j+1} \tag{26}$$

where $\mathbf{F}_{Gt}$ is any $j+1$ column of matrices in $\mathbf{L}$ as in Equation (8).

A New Form of L With Conditional Covariances

The $\mathbf{L}$ matrix can be written in terms of successive conditional covariances and within set conditional $\mathbf{L}_{jj}$ matrices. See, for example, Equations (12), (19), and (25). Applying this type of equations to the $\mathbf{L}$ matrix yields

$$\mathbf{L} = \begin{bmatrix} \mathbf{L}_{pp} & & & & & \\ \mathbf{L}_{pp}\left[\mathbf{C}_{pp}^{-1}\mathbf{C}_{ps}\right] & \mathbf{L}_{ss} & & & & \\ \vdots & \vdots & & & & \\ \mathbf{L}_{pp}\left[\mathbf{C}_{pp}^{-1}\mathbf{C}_{pt}\right] & \mathbf{L}_{ss}\left[\boldsymbol{\xi}_{ss}^{-1}\boldsymbol{\xi}_{st}\right] & \cdots & \mathbf{L}_{tt} & & \\ \mathbf{L}_{pp}\left[\mathbf{C}_{pp}^{-1}\mathbf{C}_{pv}\right] & \mathbf{L}_{ss}\left[\boldsymbol{\xi}_{ss}^{-1}\boldsymbol{\xi}_{sv}\right] & \cdots & \mathbf{L}_{tt}\left[\left[\boldsymbol{\xi}_{tt}^{j}\right]^{-1}\boldsymbol{\xi}_{tv}^{j}\right] & \mathbf{L}_{vv} & \\ \vdots & \vdots & & \vdots & \vdots & \\ \mathbf{L}_{pp}\left[\mathbf{C}_{pp}^{-1}\mathbf{C}_{pm}\right] & \mathbf{L}_{ss}\left[\boldsymbol{\xi}_{ss}^{-1}\boldsymbol{\xi}_{sm}\right] & \cdots & \mathbf{L}_{tt}\left[\left[\boldsymbol{\xi}_{tt}^{j}\right]^{-1}\boldsymbol{\xi}_{tv}^{j}\right] & \mathbf{L}_{vv}\left[\left[\boldsymbol{\xi}_{vv}^{j+1}\right]^{-1}\boldsymbol{\xi}_{vm}^{j+1}\right] & \cdots \mathbf{L}_{mm} \end{bmatrix} \tag{27}$$

Equation (27) is a very practical way of computing $\mathbf{L}$ matrices for large covariance matrices. Note this proposition is very interesting because the computation of any $j+1$ column only needs the knowledge of the previous j column. This means that simultaneous computations with the whole covariance matrix $\mathbf{C}_G$ are not necessary. Equation (27) can also be written as

$$\begin{aligned} \mathbf{L} = \big\lfloor \mathbf{L}_{pp}\left(\mathbf{C}_{pp}^{-1}\mathbf{C}_{pG}\right) \mathbf{L}_{ss}\left(\boldsymbol{\xi}_{ss}^{-1}\boldsymbol{\xi}_{sG}\right) \cdots \mathbf{L}_{tt}\left(\left[\boldsymbol{\xi}_{tt}^{j}\right]^{-1}\boldsymbol{\xi}_{tG}^{j}\right) \\ \mathbf{L}_{vv}\left(\left[\boldsymbol{\xi}_{vv}^{j+1}\right]^{-1}\boldsymbol{\xi}_{vG}^{j+1}\right) \cdots \mathbf{L}_{mm} \big\rfloor \end{aligned} \tag{28}$$

We have found that Equation (28) is a successive $\mathbf{LU}$ decomposition. Note that the kriging weights in each term, this leads to

$$\mathbf{L} = \left[\mathbf{L}_{pp}\left(\boldsymbol{\Lambda}_{pG}^{1}\right) \mathbf{L}_{ss}\left(\boldsymbol{\Lambda}_{sG}^{2}\right) \cdots \mathbf{L}_{tt}\left(\boldsymbol{\Lambda}_{tG}^{j}\right) \mathbf{L}_{vv}\left(\boldsymbol{\Lambda}_{vG}^{j+1}\right) \cdots \mathbf{L}_{mm}\right] \tag{29}$$

The advantage for computation is in the property of using just the previous column for updating the conditioning, for example, just the third column is used to update successive covariances in the fourth column and so forth as in Equation (24).

Conditional Expression of Simulation by LU

The conditional $\mathbf{L}_{jj}$, computed from updated covariances for the residual random variables within a set $\boldsymbol{\Omega}_j$ in Equation (27), are considered separately. The residuals between the "true" $\mathbf{z}_j$ and kriging estimates $\hat{\mathbf{z}}_j$ are computed as $\Delta\mathbf{z}_j = \mathbf{L}_{jj}\mathbf{w}_j$. Then, simulation by **LU** decomposition can be written as

$$\begin{bmatrix} \mathbf{I}_{pp} & & & & & \\ [\mathbf{C}_{pp}^{-1}\mathbf{C}_{ps}]^{\mathrm{T}} & \mathbf{I}_{ss} & & & & \\ \vdots & \vdots & & & & \\ [\mathbf{C}_{pp}^{-1}\mathbf{C}_{pt}]^{\mathrm{T}} & [\boldsymbol{\xi}_{ss}^{-1}\boldsymbol{\xi}_{st}]^{\mathrm{T}} \cdots & \mathbf{I}_{tt} & & & \\ [\mathbf{C}_{pp}^{-1}\mathbf{C}_{pv}]^{\mathrm{T}} & [\boldsymbol{\xi}_{ss}^{-1}\boldsymbol{\xi}_{sv}]^{\mathrm{T}} \cdots & [[\boldsymbol{\xi}_{tt}^{j}]^{-1}\boldsymbol{\xi}_{tv}^{j}]^{\mathrm{T}} & \mathbf{I}_{vv} & & \\ \vdots & \vdots & \vdots & \vdots & & \\ [\mathbf{C}_{pp}^{-1}\mathbf{C}_{pm}]^{\mathrm{T}} & [\boldsymbol{\xi}_{ss}^{-1}\boldsymbol{\xi}_{sm}]^{\mathrm{T}} \cdots & [[\boldsymbol{\xi}_{tt}^{j}]^{-1}\boldsymbol{\xi}_{tm}^{j}]^{\mathrm{T}} & [[\boldsymbol{\xi}_{vv}^{j}]^{-1}\boldsymbol{\xi}_{vm}^{j}]^{\mathrm{T}} \cdots & \mathbf{I}_{mm} \end{bmatrix} \begin{bmatrix} \mathbf{z}_p \\ [\mathbf{z}_s - \hat{\mathbf{z}}_s] \\ \vdots \\ [\mathbf{z}_t - \hat{\mathbf{z}}_t] \\ \mathbf{L}_{vv}\mathbf{w}_v \\ \vdots \\ \mathbf{L}_{mm}\mathbf{w}_m \end{bmatrix} = \begin{bmatrix} \mathbf{z}_p \\ \mathbf{z}_s \\ \vdots \\ \mathbf{z}_t \\ \mathbf{z}_v \\ \vdots \\ \mathbf{z}_m \end{bmatrix} \tag{30}$$

This formulation and Equation (29) can express any simulated set as the sum of column vectors for a set of points $\boldsymbol{\Omega}_i$ as follows

$$\begin{aligned} \mathbf{z}_i = {} & \left(\boldsymbol{\Lambda}_{pi}^{1}\right)^{\mathrm{T}}\mathbf{z}_p + \left(\boldsymbol{\Lambda}_{si}^{2}\right)^{\mathrm{T}}\Delta\mathbf{z}_s + \cdots + \left(\boldsymbol{\Lambda}_{ti}^{j}\right)^{\mathrm{T}}\Delta\mathbf{z}_t \\ & + \left(\boldsymbol{\Lambda}_{vi}^{j+1}\right)^{\mathrm{T}}\mathbf{L}_{vv}\mathbf{w}_v + \cdots + \mathbf{L}_{ii}\mathbf{w}_i \end{aligned} \tag{31}$$

where $\boldsymbol{\Lambda}$ are the residual kriging weights. One recognizes that Equation (31) can be solved sequentially up to the available conditioning data $\boldsymbol{\Omega}_t$. The weights in Equation (31) can be computed from the successive minimization of the estimation variances shown in Appendix. It can be shown that the simultaneous kriging solution is equivalent to the sequential or successive kriging solution because the minimization of estimation variance by steps is independent of the order in which the data values are utilized (Vargas-Guzmán and Yeh, 1999). The complete expression in Equation (31) can not be computed simultaneously unless the updated conditional covariances are known. The matrix constructed from conditional covariance is independent of the data values and will remain unchanged if simulated locations become new data locations. As a result, updating of old realizations can be performed with conditional covariances modifying the vector of residuals as observed in Equation (30).

If all sets $\boldsymbol{\Omega}_j$ are unit sets as in sequential Gaussian method, then the kriging weights in Equation (31) are just scalar single numbers a_j or conditional spatial autocorrelations. The conditional **L** matrices reduce to a square root of the conditional variance (i.e., kriging standard deviation). We propose an approach, based on updated conditional covariances, which does not require solving the kriging

system of equations. Equation (31) becomes

$$z_i - a_p z_p - a_s \Delta z_s - \cdots a_t \Delta z_t = a_v L_{vv} w_v + \cdots L_{ii} w_i \tag{32}$$

The left-hand side of Equation (32) is the conditional mean part and the right hand side is a moving average producing the correlated simulated residual from white noise. This is analogous to ARMA models.

Extension to the Multivariate Case

The proposed approach can be extended to the multivariate case of a vector random field Z(x). Equation (30) can be modified by using multivariate conditional covariances matrices which include conditional cross-covariances.

$$\begin{bmatrix} \mathbf{I}_{pp} & & & & & \\ [\bar{\mathbf{C}}_{pp}^{-1}\bar{\mathbf{C}}_{ps}]^{\mathrm{T}} & \mathbf{I}_{ss} & & & & \\ \vdots & \vdots & & & & \\ [\bar{\mathbf{C}}_{pp}^{-1}\bar{\mathbf{C}}_{pt}]^{\mathrm{T}} & [\bar{\boldsymbol{\xi}}_{ss}^{-1}\bar{\boldsymbol{\xi}}_{st}]^{\mathrm{T}} \cdots & \mathbf{I}_{tt} & & & \\ [\bar{\mathbf{C}}_{pp}^{-1}\bar{\mathbf{C}}_{pv}]^{\mathrm{T}} & [\bar{\boldsymbol{\xi}}_{ss}^{-1}\bar{\boldsymbol{\xi}}_{sv}]^{\mathrm{T}} \cdots & [[\bar{\boldsymbol{\xi}}_{tt}^{j}]^{-1}\bar{\boldsymbol{\xi}}_{tv}^{j}]^{\mathrm{T}} & \mathbf{I}_{vv} & & \\ \vdots & \vdots & \vdots & \vdots & & \\ [\bar{\mathbf{C}}_{pp}^{-1}\bar{\mathbf{C}}_{pm}]^{\mathrm{T}} & [\bar{\boldsymbol{\xi}}_{ss}^{-1}\bar{\boldsymbol{\xi}}_{sm}]^{\mathrm{T}} \cdots & [[\bar{\boldsymbol{\xi}}_{tt}^{j}]^{-1}\bar{\boldsymbol{\xi}}_{tm}^{j}]^{\mathrm{T}} & [[\bar{\boldsymbol{\xi}}_{vv}^{j+1}]^{-1}\bar{\boldsymbol{\xi}}_{vm}^{j+1}]^{\mathrm{T}} \cdots & \mathbf{I}_{mm} \end{bmatrix} \begin{bmatrix} \mathbf{Z}_p \\ [\mathbf{Z}_s - \hat{\mathbf{Z}}_s] \\ \vdots \\ [\mathbf{Z}_t - \hat{\mathbf{Z}}_t] \\ \bar{\mathbf{L}}_{vv}\mathbf{W}_v \\ \vdots \\ \bar{\mathbf{L}}_{mm}\mathbf{W}_m \end{bmatrix} = \begin{bmatrix} \mathbf{Z}_p \\ \mathbf{Z}_s \\ \vdots \\ \mathbf{Z}_u \\ \mathbf{Z}_v \\ \vdots \\ \mathbf{Z}_m \end{bmatrix} \tag{33}$$

where $\mathbf{W}$ are matrices of white noise. The matrices of residuals are $\Delta\mathbf{Z}_i = \mathbf{Z}_i - \hat{\mathbf{Z}}_i$. Note the notation of bars on the covariance and $\mathbf{L}$ matrices in Equation (33) indicates they are multivariate. The residuals are also made of matrices that have one column for each attribute. The results of the simulation are also matrices of multivariate realizations. Equation (33) formulates a multivariate conditional simulation by residuals that can be solved using successive cokriging (Vargas-Guzmán and Yeh, 1999) to update conditional covariances and cross-covariances for residuals at each step of the approach. At any step, a matrix of cokriging weights $\bar{\boldsymbol{\Lambda}}_{tj}^{j+1}$ to estimate residuals for the set $\boldsymbol{\Omega}_j$ from residuals of a set $\boldsymbol{\Omega}_s$ is computed as

$$\bar{\boldsymbol{\xi}}_{tt}^{j+1}\bar{\boldsymbol{\Lambda}}_{tj}^{j+1} = \bar{\boldsymbol{\xi}}_{tj}^{j+1} \tag{34}$$

The conditional cross-covariances in Equation (34) are the off-diagonal terms of the conditional multivariate covariance matrices and may not be symmetric. For example, between two sets of locations $\boldsymbol{\Omega}_u$ and $\boldsymbol{\Omega}_v$ the matrix of updated

conditional multivariate covariance for two random fields $Z(x)$ and $Y(x)$ at step $j+1$ is written in terms of j step conditional covariance and the cokriging weights as follows

$$\bar{\boldsymbol{\xi}}_{uv}^{j+1} = \begin{bmatrix} \boldsymbol{\xi}_{Z_u Z_v}^{j} & \boldsymbol{\xi}_{Z_u Y_v}^{j} \\ \boldsymbol{\xi}_{Y_u Z_v}^{j} & \boldsymbol{\xi}_{Y_u Y_v}^{j} \end{bmatrix} - \begin{bmatrix} \boldsymbol{\Lambda}_{Z_d Z_u}^{j} & \boldsymbol{\Lambda}_{Z_d Y_u}^{j} \\ \boldsymbol{\Lambda}_{Y_d Z_u}^{j} & \boldsymbol{\Lambda}_{Y_d Y_u}^{j} \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \boldsymbol{\xi}_{Z_d Z_v}^{j} & \boldsymbol{\xi}_{Z_d Y_v}^{j} \\ \boldsymbol{\xi}_{Y_d Z_v}^{j} & \boldsymbol{\xi}_{Y_d Y_v}^{j} \end{bmatrix} \tag{35}$$

Then, a realization of the joint simulation can be computed as the sum of matrices such as

$$\begin{aligned} \mathbf{Z}_i = {} & \left(\bar{\boldsymbol{\Lambda}}_{pi}^{1}\right)^{\mathrm{T}} \mathbf{Z}_p + \left(\bar{\boldsymbol{\Lambda}}_{si}^{2}\right)^{\mathrm{T}} \Delta \mathbf{Z}_s + \cdots + \left(\bar{\boldsymbol{\Lambda}}_{ti}^{j}\right)^{\mathrm{T}} \Delta \mathbf{Z}_t \\ & + \left(\bar{\boldsymbol{\Lambda}}_{vi}^{j+1}\right)^{\mathrm{T}} \bar{\mathbf{L}}_{vv} \mathbf{W}_v + \cdots + \bar{\mathbf{L}}_{ii} \mathbf{W}_i \end{aligned} \tag{36}$$

Equation (36) is computed sequentially as Equation (31) in the univariate case.

THE STEP BY STEP ALGORITHM

Univariate Case

Equation (31) summarizes the CSSR method. The algorithm follows the next steps.

(a) A conditioning data set $\boldsymbol{\Omega}_d$ is partitioned into a number of small subsets $\boldsymbol{\Omega}_d = \boldsymbol{\Omega}_p \cup \boldsymbol{\Omega}_s \cup \cdots \cup \boldsymbol{\Omega}_t$.

(b) Equation (26) is utilized to estimate the entire correlated field with a small subset of data at each step. In practice, residual conditional covariances have values significantly larger than zero only within a local neighborhood.

(c) If conditioning data are still available continue successive estimation of residuals in (b), otherwise continue with (d). Note that previously used data subsets are automatically removed because a conditional covariance between any location and older data location is zero.

(d) A subset of locations to be simulated is randomly chosen from $\boldsymbol{\Omega}_g = \boldsymbol{\Omega}_v \cup \cdots \cup \boldsymbol{\Omega}_m$ and may be one of the next alternatives
- The subset of point locations is spread randomly throughout the domain;
- The subset of point locations are within a cluster or block for simulation.

(e) Update conditional covariances between the subset and the whole domain and within the subset, Equation (24).

(f) A vector $\mathbf{w}_j$ of pure white noise is drawn from a cdf normal (0,1).
(g) The small matrix of residual conditional covariances computed within the domain subset is decomposed by **LU**.
(h) The small conditional $\mathbf{L}_{jj}$ matrix is multiplied by the vector of pure white noise and a vector of residuals is obtained as $\Delta\mathbf{z}_j = \mathbf{L}_{jj}\mathbf{w}_j$.
(i) The vector of residuals is added to the previous estimates and stored.
(j) Go to (b) and use the vector of residuals $\Delta\mathbf{z}_j$ to update estimates everywhere.

This algorithm becomes even simpler than the above if the simulated subset reduces to a single point at a time, as can be seen from Equation (32). However, successive simulations using subsets can improve computational speed. The latter is further enhanced by the successive approach, as it only requires that the current j step conditional covariances are stored in memory until right after they are updated at the $j+1$ step. The updating of covariances only requires matrix products.

Multivariate Case

The multivariate simulation implies modifications of the algorithm proposed for the univariate case. Every covariance matrix must be changed by a multivariate matrix of covariances. Data are partitioned into multivariate groups of point locations but also may split into attributes and locations. The introduction of several attributes leads to successive cokriging which is used instead of kriging. The residuals are matrices computed in a similar way to the univariate case and the pure white noise is also in the form of matrices where each column is allocated to one attribute. Computation of conditional **L** matrices is made from multivariate conditional covariances matrices.

SUMMARY AND CONCLUSIONS

A novel approach to conditional simulation based on the **LU** decomposition using conditional covariances is presented in this paper. The approach involves a partitioning of the **L** matrix by columns using conditional covariance terms that allow the **LU** decomposition of a covariance matrix to be performed in a successive fashion. Conditional covariances from the previous column are used to compute the terms of the next column in the **L** matrix. It has been shown that the complete **L** matrix can be computed as the product of kriging weights for residuals and conditional $\mathbf{L}_{jj}$ matrices obtained from updated residual conditional covariances within a group of locations, as is shown in Equations (26) and (27). When the simulation process is carried out for a single node at a time, the conditional $\mathbf{L}_{jj}$

is reduced to the square root of the kriging variance, thus removing the need for matrix inversion calculations in the updating of residual estimates. This new approach allows for partitioned computation of large **L** matrices and successive simulation by residuals.

The complete **L** matrix can be computed by following a known sequence, or pathway, of nodes to be simulated. The computation of **L** is done by columns or a group of columns at a time, using conditional covariances from the previous group of columns only. The order of the columns in the simultaneous **LU** decomposition carries on information about the pathway being followed by the simulation process. Hence the **L** matrices can be considered different for different realizations.

The proposed CSSR method eliminates the computational upper limit of the traditional LU decomposition and allows for any size simulations, thus providing an alternative option for the simulation of very large Gaussian random fields. While retaining the relative simplicity and efficiency attributed to traditional **LU**-based approaches, successive conditional simulation by residuals has the computationally attractive feature of being able to simulate several nodes at a time.

Although the **L** matrix computed by columns does not need to be kept, a stored **L** matrix can easily facilitate the conditional updating of the realizations. If a set of locations Ω_j within the simulated domain becomes known, updating of the realizations is performed using a product of **L** by a modified vector of residuals. The use of future updating requires knowledge of the future sequence of data locations that may become available and assumes that the covariance model remains invariant as additional data are included.

There are several areas of possible future research based on the developments of this paper. This may include a detailed study of computational efficiency of the proposed successive simulation algorithm. The extension of the approach to the simulation of non-Gaussian random fields, the direct block support scale simulation and space time simulations also deserve research. In addition, the theoretical contribution of partitioning of the **L** matrix by columns opens possibilities for research on the successive decomposition of matrices of higher-order moments for nonlinear and multipoint geostatistics.

REFERENCES

Alabert, F., 1987, The practice of fast conditional simulations through the LU decomposition of the covariance matrix: Math. Geol., v. 19, no. 5, p. 368–386.

Davis, M. W., 1987a, Production of conditional simulations via the LU triangular decomposition of the covariance matrix: Math. Geol., v. 19, no. 2, p. 91–98.

Davis, M. W., 1987b, Generating large stochastic simulation—The matrix polynomial approximation method: Math. Geol., v. 19, no. 2, p. 99–107.

Deutsch, C. V., and Journel, A. G., 1998, GSLIB geostatistical software library and user's guide: Oxford University Press, New York, 369 p.

Dietrich, C. R., and Newsam, G. N., 1995, Efficient generation of conditional simulations by Chebyshev matrix polynomial approximations to symmetric square root of the covariance matrix: Math. Geol., v. 27, no. 2, p. 207–228.

Dowd, P. A., and Saraç, C., 1993, An extension of the LU decomposition method of simulation, *in* Armstrong, M., and Dowd, P. A., eds., Geostatistical simulations: Kluwer Academic, London, p. 23–36.

Isaaks, E. H., 1990, The application of Monte Carlo methods to the analysis of spatially correlated data: Ph.D. thesis, Stanford University.

Johnson, M., 1987, Multivariate statistical simulation: Wiley, New York, 212 p.

Journel, A. G., 1994, Modelling uncertainly: Some conceptual thoughts, *in* Dimitrakopoulos, R., ed., Geostatistics for the next century: Kluwer Academic, Dordtrecht, The Netherlands, p. 30–43.

Vargas-Guzmán, J. A., and Yeh, T.-C. J., 1999, Sequential kriging and cokriging, two powerful geostatistical approaches: Stochastic Environ. Res. Risk Assess., v. 13, no. 6, p. 416–435.

APPENDIX: SUCCESSIVE OR SEQUENTIAL KRIGING

Successive or sequential kriging (Vargas-Guzmán and Yeh, 1999) is recalled in several parts of this paper. The term successive is used to avoid confusion with sequential simulations. Sequential kriging is the estimator that minimizes the estimation variance by steps rather than simultaneously. The data set is split into several subsets, and each subset may be a single datum location. At some step $j+1$ of the estimation process, the utilized sample data vector is considered partitioned in old $\mathbf{z}_p$ and new samples $\mathbf{z}_s$. This is

$$\mathbf{z}_d = \begin{bmatrix} \mathbf{z}_p \\ \mathbf{z}_s \end{bmatrix} \tag{A1}$$

Then, sequential kriging consists in using new sample locations to update estimates previously made with the old samples at step j. So at each step, new sample data locations become available. The simplified version of the estimator is

$$\hat{\mathbf{z}}_o^{j+1} = \left[\boldsymbol{\Lambda}_{po}^{j}\right]^{\mathrm{T}} \mathbf{z}_p + \left[\boldsymbol{\Lambda}_{so}^{j+1}\right]^{\mathrm{T}} \left(\mathbf{z}_s - \left[\boldsymbol{\Lambda}_{ps}^{j}\right]^{\mathrm{T}} \mathbf{z}_p\right) \tag{A2}$$

where $\boldsymbol{\Lambda}_{po}^{j}$ are weights for estimation at a point with the old data and $\boldsymbol{\Lambda}_{so}^{j+1}$ are weights for updating the estimate with residuals. The kriging estimations are performed everywhere in the domain without need of local neighborhood or other mechanisms.

From the second to the last step, sequential kriging is performed using successively updated conditional covariances and residuals. The process continues until no more data are available. By expressing the estimation variance in terms of the successive weights of Equation (A2), Vargas-Guzmán and Yeh (1999) show

a sequential set of kriging equations in terms of updated conditional covariances to be

$$\left[\boldsymbol{\xi}_{tt}^{j} - \left(\boldsymbol{\xi}_{ts}^{j}\right)^{\mathrm{T}}\left(\boldsymbol{\xi}_{ss}^{j}\right)^{-1}\boldsymbol{\xi}_{st}^{j}\right]\boldsymbol{\Lambda}_{to}^{j+1} = \left[\boldsymbol{\xi}_{to}^{j} - \left(\boldsymbol{\xi}_{ts}^{j}\right)^{\mathrm{T}}\left(\xi_{ss}^{j}\right)^{-1}\boldsymbol{\xi}_{so}^{j}\right] \tag{A3}$$

These updated conditional covariances are the same like those we have found in the **L** matrix, see Equation (24).