Editorial Manager(tm) for Mathematical Geology
Manuscript Draft

Manuscript Number: MATG188

Title: FFT-based Algorithms for Kriging

Article Type: Original Research

Section/Category:

Keywords: Fast Fourier Transform; Geostatistical Estimation; Spectral Methods; Superfast Toeplitz Solver

Corresponding Author: Dipl.-Ing. Jochen Fritz,

Corresponding Author's Institution: Universität Stuttgart

First Author: Jochen Fritz

Order of Authors: Jochen Fritz; Insa Neuweiler, Dr.rer.nat.; Wolfgang Nowak, Dr.-Ing.

Manuscript Region of Origin:

Abstract: Computational power and storage capacities often pose heavy limitations to the size of the problem able to be addressed in Kriging. For estimation on regular grids and the generalized case of an uncertain and spatially varying mean, we compile a toolbox of FFT-based (spectral) methods for Kriging that is highly efficient in storage and computational complexity. The general theme is to apply, improve or extend existing FFT-based algorithms for basic operations on covariance matrices which apply when covariance matrices have Toeplitz structure. The discussed FFT-based algorithms are easily applicable for the case of regular grids. In case of irregularly scattered data, we trace the problem back to sparse but regular finer grids of measurements. We also present several fast approximations for the estimation variance of Kriged fields that are asymptotically exact for certain limiting cases. The computational efficiency and reduction of storage requirements over existing Kriging algorithms are discussed and demonstrated in test cases.

FFT-based Algorithms for Kriging*

J.Fritz†, I. Neuweiler‡and W.Nowak§

September 11, 2007

Corresponding Author:
Jochen Fritz
Institute of Hydraulic Engineering
Pfaffenwaldring 61
70569 Stuttgart
Germany
jochen.fritz@iws.uni-stuttgart.de
Phone: +49 (0)711 / 685-64667
Fax: +49 (0)711 / 685 - 60430

ABSTRACT

Computational power and storage capacities often pose heavy limitations to the size of the problem able to be addressed in Kriging. For estimation on regular grids and the generalized case of an uncertain and spatially varying mean, we compile a toolbox of FFT-based (spectral) methods for Kriging that is highly efficient in storage and computational complexity. The general theme is to apply, improve or extend existing FFT-based algorithms

*Received ; accepted

†Department of Hydromechanics and Modeling of Hydrosystems, Institute of Hydraulic Engineering, Universität Stuttgart, Stuttgart, Germany, Tel. +49 (0)711 685-64667, Jochen.Fritz@iws.uni-stuttgart.de

‡Department of Hydromechanics and Modeling of Hydrosystems, Institute of Hydraulic Engineering, Universität Stuttgart, Stuttgart, Germany, Tel. +49 (0)711 685-69165, Insa.Neuweiler@iws.uni-stuttgart.de

§Department of Hydromechanics and Modeling of Hydrosystems, Institute of Hydraulic Engineering, Universität Stuttgart, Stuttgart, Germany, Tel. +49 (0)711 685-68218, Wolfgang.Nowak@iws.uni-stuttgart.de

for basic operations on covariance matrices which apply when covariance matrices have Toeplitz structure. The discussed FFT-based algorithms are easily applicable for the case of regular grids. In case of irregularly scattered data, we trace the problem back to sparse but regular finer grids of measurements. We also present several fast approximations for the estimation variance of Kriged fields that are asymptotically exact for certain limiting cases. The computational efficiency and reduction of storage requirements over existing Kriging algorithms are discussed and demonstrated in test cases.

KEY WORDS: Fast Fourier Transform, Geostatistical Estimation, Spectral Methods

INTRODUCTION

Spatially distributed quantities such as rainfall intensities, contaminant concentrations or hydraulic conductivities are frequently interpolated between scattered measurements by Kriging. Especially when considering large data sets of measurements, Kriging can lead to systems of equations which are far beyond the storage capacities and computational power of contemporary desktop computers. The motivation of this work is to make Kriging fast and the required storage capacities low, permitting the solution of very large problems on small computers.

In most standard forms of Kriging, there are three computationally most demanding tasks. These are (1) to solve an $m \times m$ system of equations that involves the auto-covariance matrix of the measurements to obtain the Kriging weights, (2) to perform a superposition with the cross-covariance function between measurements and unknowns, weighted by the Kriging weights in order to obtain the values at the points of estimation, and (3) to repeat this procedure once for the estimator and m times for the estimation variance, where m is the number of measurements.

Nearest neighborhood Kriging avoids the solution of large systems of equations by only considering measurements within a certain radius around every point of estimation. This requires the setting up and solution of as many systems of equations as the number of different neighborhoods, and may be very time consuming (e.g.Kitanidis, 1997). Also, an

implicit assumption on a moving-average type of mean value has to be made since the mean value cannot be estimated globally.

Pegram (2004) published an efficient FFT-based Kriging method. He solves the Kriging equations using an FFT-based algorithm called Iterative Constrained Deconvolution (ICD). It is restricted to the cases of either known or unknown constant mean. For the latter case, the sample mean of the measurements is used to estimate the unknown mean value of the field. The estimation variance is approximated with only one system of equations to be solved, assuming that the measurements are almost uncorrelated. We will demonstrate that the ICD algorithm, being formally identical to a non-preconditioned steepest descent with an empirically chosen step-size coefficient, can be replaced by a more efficient PCG-based algorithm. Also, we will show how to generalize Kriging in the FFT-context to the case of uncertain mean, including spatial trends.

We exploit the fact that, in most cases, the auto-covariance function of the unknowns is assumed to be second-order stationary or at least intrinsic (e.g, Kitanidis 1997), and the points of estimation lie on a regular and equispaced grid. This setup leads to covariance matrices with symmetric Toeplitz structure (Golub and van Loan 1996). The Toeplitz structure can be exploited in terms of storage, because only the first column of the matrix has to be stored (Zimmerman 1989). The product of a Toeplitz matrix with a vector is the same as discrete convolution of a vector with a corresponding (covariance) function. This convolution can be performed quickly using the FFT algorithm for convolution (e.g., van Loan 1992). We show how to perform superposition based on this convolution algorithm in order to speed up task (2) of evaluating the values at the points of estimation in Kriging. With this technique, the associated computational costs can be reduced from the order of mn to the order of $n \log_2 n$, where n is the number of estimation points. This is an advantage for medium and large numbers of measurements ($m > \log_2 n$).

The more fundamental problem, task (1), is to solve the $m \times m$ system of Kriging equations with the auto-covariance matrix of the measurements. If the measurements lie on a regular grid and the field is second-order stationary, their auto-covariance matrix is again a Toeplitz matrix. Per definition of covariance functions, the Toeplitz matrix is symmetric

and positive-definite. Solving a Toeplitz system has been the subject of many studies in the signal processing community (e.g. Gallivan et al. 1996, Kailath and Sayed 1999, Barel et al. 2001). The iterative Toeplitz solver that we find the most promising within the Kriging context is the FFT-based Preconditioned Conjugate Gradient (FFT-PCG) method with circulant preconditioners. Applying this algorithm has the same prerequisites as convolution via FFT, plus positive-definiteness of the involved Toeplitz matrix. We refer to Shewchuk (1994) for a graphical review of the PCG algorithm and to Chan and Ng (1996) for a comprehensive review on the use of circulant preconditioners therein. Using the FFT-PCG algorithm, the Kriging system can be solved with computational effort in the order of $m \log_2 m$ instead of m^3 (depending on the conventional solver used).

In case the measurements are irregularly spaced, the Toeplitz structure of the auto-covariance matrix of measurements is lost. For this case, we present an extension of the FFT-PCG algorithm to irregular grids. This extension is similar to the idea used by Pegram (2004) in the ICD algorithm, but embedded in a more powerful iterative solver.

Most intrinsic random space functions can be traced back to special cases of second-order stationary fluctuations about a spatial varying mean with known base functions (i.e., trends) but unknown or uncertain coefficients of the base functions (compare Nowak and Cirpka 2004, Nowak and Cirpka 2006, Kitanidis 1993). In order to be as general as possible and include such intrinsic cases, we use the generalized form of Kriging for the case of a spatially variable and uncertain mean. This case also includes the cases of known and unknown mean as limiting cases.

The remaining problem is that task (1) and (2) need to be performed m times in order to obtain the estimation variance of Kriging. We show how to reduce the factor of m in several levels of trade-off versus accuracy. These approximations are asymptotically exact for certain special cases. The fastest options require exactly one additional solution, using the approximation for almost uncorrelated measurements as suggested by Pegram (2004), or by neglecting boundary effects on large regular grids of measurements. At the end of this study, we demonstrate that these methods allow to handle huge Kriging problems even on ordinary desktop computers.

KRIGING WITH UNCERTAIN MEAN

Based on the function-estimate form of Kriging with unknown mean (e.g., Kitanidis 1997 and Kitanidis 1996), Nowak and Cirpka (2004) developed the generalization of Kriging to the case of uncertain mean. Although their original publication dealt with non-linear cokriging-like cases, Kriging is obtained from the described method by using the same covariance function among observed and estimated quantities. The following is a brief summary of the method.

Let $\mathbf{s}$ be an $n \times 1$ multi-Gaussian vector of unknowns (sometimes also called target point values in regression-like problems) with expectation $E[\mathbf{s}] = \mathbf{X}\boldsymbol{\beta}$ and covariance $Cov[\mathbf{s}|\boldsymbol{\beta}] = \mathbf{Q_{ss}}$, where $\mathbf{X}$ denotes the $[n \times p]$ matrix of discrete base functions and $\boldsymbol{\beta}$ the $[p \times 1]$ vector of trend coefficients. For a spatially constant mean of $\mathbf{s}$, $\mathbf{X}$ is a $n \times 1$ vector with unit entries and $\boldsymbol{\beta}$ is the actual value of the mean. When adding trends, $\mathbf{X}$ is extended by new columns for each additional trend term, with the according coefficient appended to $\boldsymbol{\beta}$. For the case of uncertain mean, the trend coefficients are again set to be Gaussian variables with mean $\boldsymbol{\beta}^*$ and variance $\mathbf{Q}_{\boldsymbol{\beta}\boldsymbol{\beta}}$. For given values of $\mathbf{s}$, the distribution of $\boldsymbol{\beta}$ is again Gaussian with conditional mean $E[\beta|\mathbf{s}] = \hat{\boldsymbol{\beta}}$ and conditional covariance $Q_{\boldsymbol{\beta}\boldsymbol{\beta}|s}$.

Consider further $\mathbf{Y}$ an $m \times 1$ vector of measurements (sometimes also called control point values in regression-like problems) which are sampled from $\mathbf{s}$ according to

$$\mathbf{Y} = \mathbf{H}\mathbf{s} + \mathbf{r}\,. \tag{1}$$

Here, $\mathbf{H}$ is a $m \times n$ sampling defined as

$$\mathbf{H}_{i,j} = \begin{cases} 1 & \text{for } \mathbf{x}_i = \mathbf{x}_j \\ 0 & \text{otherwise} \end{cases}\,, \tag{2}$$

with $\mathbf{x}_i$ the coordinates of the i-th measurement location and $\mathbf{x}_j$ the coordinates of the j-th estimation point, and $\mathbf{r}$ is an $m \times 1$ vector of random measurement errors. The

measurement errors have zero mean and $m \times m$ covariance matrix $\mathbf{R}$, typically a scalar matrix (i.e., a diagonal matrix with constant values along the diagonal). Interpreting the sampling matrix $\mathbf{H}$ as a sensitivity matrix in linear error propagation, the following identities hold:

$$\mathbf{Q_{ys}} = \mathbf{H}\mathbf{Q_{ss}} \tag{3}$$

$$\mathbf{Q_{sy}} = \mathbf{Q_{ss}}\mathbf{H}^T \tag{4}$$

$$\mathbf{Q_{yy}} = \mathbf{H}\mathbf{Q_{ss}}\mathbf{H}^T + \mathbf{R}\ , \tag{5}$$

where $\mathbf{Q_{sy}}$ is the $n \times m$ cross-covariance matrix between the unknowns $\mathbf{s}$ and measurements $\mathbf{Y}$, and $\mathbf{Q_{yy}}$ is the $m \times m$ auto-covariance matrix of the measurements. In this notation, the Kriging estimate $\hat{\mathbf{s}}$ is given by

$$\hat{\mathbf{s}} = \begin{bmatrix} \mathbf{Q_{ys}} \\ \mathbf{X}^T \end{bmatrix}^T \begin{bmatrix} \boldsymbol{\xi} \\ \widehat{\boldsymbol{\beta}} \end{bmatrix}\ . \tag{6}$$

The $m \times 1$ vector of Kriging weights $\boldsymbol{\xi}$ (also called reciprocal data by Pegram 2004) and the $p \times 1$ vector of trend coefficients $\hat{\boldsymbol{\beta}}$ are taken from the solution of the Kriging system

$$\begin{bmatrix} \mathbf{Q_{yy}} & \mathbf{HX} \\ (\mathbf{HX})^T & -\mathbf{Q^{-1}_{\beta\beta}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi} \\ \widehat{\boldsymbol{\beta}} \end{bmatrix} = \begin{bmatrix} \mathbf{Y} \\ -\mathbf{Q^{-1}_{\beta\beta}}\boldsymbol{\beta}* \end{bmatrix}\ . \tag{7}$$

The associated estimation variance $\hat{\boldsymbol{\sigma}}$ is the $n \times 1$ vector on the diagonal of the conditional covariance matrix

$$\mathbf{Q_{ss|y}} = \mathbf{Q_{ss}} - \begin{bmatrix} \mathbf{Q_{ys}} \\ \mathbf{X}^T \end{bmatrix}^{\mathbf{T}} \begin{bmatrix} \mathbf{Q_{yy}} & \mathbf{HX} \\ (\mathbf{HX})^T & -\mathbf{Q^{-1}_{\beta\beta}} \end{bmatrix}^{-\mathbf{1}} \begin{bmatrix} \mathbf{Q_{ys}} \\ \mathbf{X}^T \end{bmatrix}\ . \tag{8}$$

The special case of unknown mean is recovered by setting $\mathbf{Q^{-1}_{\beta\beta}} = 0$ and using a vector of unit entries for $\mathbf{X}$. The case of known mean is included by setting $p = 0$, i.e., by omitting all rows and columns that refer to $\boldsymbol{\beta}$.

The covariance of $\mathbf{s}$ for uncertain $\boldsymbol{\beta}$ is $\mathbf{G_{ss}} = \mathbf{Q_{ss}} + \mathbf{X}\mathbf{Q}_{\boldsymbol{\beta}\boldsymbol{\beta}}\mathbf{X}^T$, where $\mathbf{G_{ss}}$ is a generalized covariance matrix $\mathbf{G_{ss}}$(Kitanidis 1993). If $\mathbf{Q_{ss}}$ is second-order stationary, then $\mathbf{G_{ss}}$ is at least intrinsic. By adequate choice of $\mathbf{X}$, the most common intrinsic cases can be expressed via generalized covariance matrices, which in turn can be decomposed into second-order stationary parts plus terms related to an uncertain mean value.

The expensive tasks referred to the introduction are (1) solving the Kriging system (Eq. 7) to obtain the Kriging weights, (2) perform the superposition $\mathbf{Q_{sy}}\boldsymbol{\xi}$ in Eq. (6) to evaluate the estimate, and (3) evaluate $m + p$ equivalents to task one and two to obtain the estimation variance (Eq. 8) .

TOOLBOX OF SPECTRAL METHODS

In this section, we provide and extend a collection of FFT-based methods which speed up matrix operations for so-called Toeplitz matrices. The connection between spatial estimation and Toeplitz matrices is explained in the following section. The basic trick of all FFT-based methods is that their periodic counterparts, called circulant matrices, have highly exploitable properties in the Fourier space. Their most central property is that the discrete Fourier transform of their first column equals their eigenvalues and that the eigenvectors of all circulant matrices are combined in the discrete Fourier matrix (Varga 1954). This so-called diagonalization theorem has also been proved by Trapp (1973). Hence, Toeplitz matrices are first converted to circulant matrices in a step called periodic embedding, then FFT-techniques are applied, followed by conversion back to the Toeplitz-equivalent in a step called extraction. Illustrated reviews of this procedure are provided by Kozintsev (1999) or by Nowak (2005).

Exploitable structure of covariance matrices

For finely resolved Kriging problems in larger domains, the unknowns $\mathbf{s}$ are typically discretized on a regular and equispaced grid. Under the common assumption that the unknowns $\mathbf{s}$ are statistically second order stationary (or intrinsic in such a way that an adequate choice of $\mathbf{X}$ yields $\mathbf{Q_{ss}}$ second-order stationary), $\mathbf{Q_{ss}}$ has symmetric Toeplitz

structure. In Toeplitz matrices, the entries along each diagonal have the same value. Hence, the $(i+1)$-th row equals the i-th row translated by one entry to the right, with a new element appearing on the first position on the left. In the d-dimensional case, the $\mathbf{Q_{ss}}$ has level-d block-Toeplitz structure, where the same pattern applies to nested block structures.

As a consequence, the first column of a symmetric Toeplitz matrix contains all the information. This has been shown to be computationally exploitable (Zimmerman 1989). First of all, only one column has to be stored instead of a full matrix, reducing storage requirements from n^2 to n elements. Second, many algorithms have been found that work on the first column only, or that work on the generating vectors only (e.g., Kailath and Sayed 1995).

A Toeplitz matrix with a first column in the form of $a_0, a_1, ..., a_N, ..., a_1$ is called a circulant matrix. Any Toeplitz matrix $\mathbf{Q_{ss}}$ with first column $q_0, q_1, ..., q_n$ can be embedded in a larger circulant matrix $\mathbf{Q_{ss},}_e$. The physical analog is embedding a finite domain in a larger periodic one. The easiest way to do this is to simply append the elements $q_1, \ldots q_{n-1}$ in reverse order $q_{n-1}, q_{n-2}, ..., q_1$ to the first column. For specific applications, smaller embedding sizes may be allowable, or larger ones to ensure positive-definiteness may be required. For example, the generation of random fields require the resulting circulant matrix to be non-negative, and it should be positive-definite if used together with certain iterative solvers, while convolution via FFT unaffected by the definiteness. These issues are discussed in more detail by Newsam and Dietrich (1994), Dietrich and Newsam (1997), or Nowak et al. (2003).

Embedding and Extraction

In matrix notation, embedding and extraction may be formally expressed via an $n_e \times n$ mapping matrix $\mathbf{M}$ (Cirpka and Nowak (2004)):

$$\mathbf{M} = \begin{bmatrix} \mathbf{I}_{n\times n} \\ \mathbf{0}_{(n_e-n)\times n} \end{bmatrix} \tag{9}$$

Then, embedding an $n \times 1$ vector $\mathbf{x}$ is

$$\text{embedding:} \quad \mathbf{x}^e = \mathbf{M}\mathbf{x} \tag{10}$$

and extracting from an $n_e \times 1$ vector $\mathbf{x}_e$ is

$$\text{extraction:} \quad \mathbf{x} = \mathbf{M}^T\mathbf{x}^e \tag{11}$$

The Toeplitz matrix can formally be extracted from the embedding circulant one by

$$\mathbf{Q_{ss}} \;=\; \mathbf{M}^T\mathbf{Q}_{\mathbf{ss},e}\mathbf{M}\;. \tag{12}$$

Of course, embedding and extraction are achieved via zero-padding or disregarding excess elements instead of performing actual matrix-vector products, or by using a sparse representation of $\mathbf{M}$. Throughout the remaining paper, $\mathbf{Q_{ss}}$ denotes Toeplitz-structured covariance matrices and $\mathbf{Q}_{\mathbf{ss},e}$ denotes larger circulant covariance matrices with first column $\mathbf{q}_{\mathbf{ss},e}$ that embed $\mathbf{Q_{ss}}$. Subscript e denotes vectors embedded according to Eq. (10).

Sampling and injection

Operations similar to embedding and extraction are injection and sampling. Measurement as a process of sampling from a vector has been defined in Eq. 2, and injection is the opposite operation. As formal matrix operations, sampling and injection are denoted by:

$$\text{sampling:} \quad \mathbf{x} = \mathbf{H}\mathbf{X} \tag{13}$$

$$\text{injection:} \quad \mathbf{X} = \mathbf{H}^T\mathbf{x} \tag{14}$$

Sampling is performed by reading specific values $\mathbf{x}$ from the larger vector $\mathbf{X}$. Injection is achieved by writing the values of $\mathbf{x}$ into a larger vector of initial zeros $\mathbf{X}$ at the sampling positions. Accordingly, we call $\mathbf{H}$ a sampling and $\mathbf{H}^T$ an injection matrix. Sampling and injection will be useful in the context of superposition and for the extension of spectral

methods to irregular grids.

Convolution via FFT

One of the most interesting properties of Toeplitz matrices is that multiplication with a vector is the same as a discrete convolution of the vector with the first column of a Toeplitz matrix. This simple and well-known trick is the basic building block for all further efficient spectral methods discussed in this paper.

Consider $\mathbf{x}$ an $n \times 1$ vector, $\mathbf{Q_{ss}}$ a $n \times n$ Toeplitz matrix, $\mathbf{q_{ss},}_e$ the first column of the embedding circulant matrix, and let $\mathcal{F}[\cdot]$ and $\mathcal{F}^{-1}[\cdot]$ denote the Fourier transform and its inverse, respectively. Then:

$$\mathbf{Q_{ss}x} = \mathbf{M}^T \mathcal{F}^{-1}\left[\mathcal{F}[\mathbf{Mx}] \circ \mathcal{F}[\mathbf{q_{ss},}_e]\right] \ , \tag{15}$$

where $(\cdot) \circ (\cdot)$ denotes the element-wise (Hadamard) product. Using the fast Fourier transform (FFT) or its extension to arbitrary vector length (Cooley and Tukey 1965, Frigo and Johnson 1998), Fourier transforming an $n \times 1$ vector has a computational complexity of $\mathcal{O}(n \log_2 n)$. Taking a small detour into a larger periodic system via circulant embedding, the product $\mathbf{Q_{ss}x}$ can be evaluated efficiently while only storing the first column of the matrix (e.g. van Loan 1992). The computational complexity is reduced from $\mathcal{O}(n^2)$ to $\mathcal{O}(n \log n)$ and storage requirements are reduced from $\mathcal{O}(n^2)$ to $\mathcal{O}(n)$. The storage requirements for standard evaluation of $\mathbf{Q_{ss}x}$ may also be reduced to $\mathcal{O}(n)$ by n successive additions of the shifted integral kernel of convolution without storing more than the current shifted version. The shifting procedure, however, also requires an embedding procedure and the same number of memory access operations as the explicit brute-force matrix product. The same FFT-based procedure and some extensions are used for evaluating auto- and cross-covariance matrices of dependent quantities in the context of cokriging and geostatistical inverse modeling in Nowak et al. (2003)).

Figure 1 (solid line and bold dash-dotted line) shows a comparison of CPU time and storage requirements depending on vector length n on a contemporary desktop computer

(i386, 2.8GHz Intel Xeon dual-core, 2GB RAM, Suse Linux 9.2, implemented in MATLAB R2006b). Although convolution via FFT has a certain overhead for embedding, it is faster even for very small problems since only n instead of n^2 elements of $\mathbf{Q_{ss}}$ need to be generated and handled in memory. The costs of conventional convolution are included as bold dash-dotted line for $n/m = 1$, while the other ratios of n over m refer to the superposition problem in the following section. Convolution via FFT is faster by almost three orders of magnitude unless for very small problems, where lower orders of computational complexity play a role. The X-marked circles indicate memory overflow. For simple conventional superposition, where n^2 elements need to be stored, storage is the main restriction to the problem size. Successive shifting of the integral kernel remedies this restriction, but does not save computational costs, resulting in the straight-line extension of the standard methods beyond the point of memory overflow.

Superposition via FFT

Evaluating the term $\mathbf{Q_{sy}}\boldsymbol{\xi}$ Eq. (6) is identical to a $\boldsymbol{\xi}$-weighted superposition of the auto-covariance of the unknowns, where the individual offset between the single superimposed terms is given by the measurement locations. In a formal step which at first looks counter-productive, we show how to trace this operation back to a convolution using Eq. (4):

$$\mathbf{Q_{sy}}\boldsymbol{\xi} = \mathbf{Q_{ss}}\mathbf{H}^T\boldsymbol{\xi} = \mathbf{Q_{ss}}\underbrace{\left(\mathbf{H}^T\boldsymbol{\xi}\right)}_{\boldsymbol{\Xi}} . \tag{16}$$

The parentheses on the right-hand side of this equation points out that we first inject the $m \times 1$ vector $\boldsymbol{\xi}$ into an $n \times 1$ vector $\boldsymbol{\Xi}$. The remaining matrix-vector product $\mathbf{Q_{ss}}\boldsymbol{\Xi}$ is again a convolution, so that:

$$\mathbf{Q_{sy}}\boldsymbol{\xi} = \mathbf{M}^T\mathcal{F}^{-1}\left[\mathcal{F}\left[\mathbf{M}\mathbf{H}^T\boldsymbol{\xi}\right] \circ \mathcal{F}\left[\mathbf{q_{ss},}_e\right]\right] , \tag{17}$$

in which we combine the injection and embedding of $\boldsymbol{\xi}$ into one operation. At a computational complexity of $\mathcal{O}\left(n \log n\right)$, this scheme is faster than direct superposition for $m > \log n$, i.e., for relatively large numbers of measurements. We also avoid the effort of

cutting/pasting /shifting the covariance function to different locations prior to summation or, even worse, to store all of these shifted functions simultaneously. Instead, only the first column $\mathbf{q_{ss},}_e$ needs to be stored, keeping the storage requirements at $\mathcal{O}(n)$ instead of $\mathcal{O}(nm)$.

The resulting computational effort and storage requirements are shown in Figure 1 for different values of n/m. In spite of its overhead and independence of m, the FFT-based algorithm is faster than the standard method in the explored range of n/m. The largest speedup, nearly three orders of magnitude compared to conventional superposition, is achieved for high numbers m of injections relative to n.

Efficient solution of Toeplitz systems

Eqs. (7) and (8) require the solutions of systems using the $m \times m$ auto-covariance matrix of the measurements, $\mathbf{Q_{yy}}$. If the measurements are arranged on a regular grid, $\mathbf{Q_{yy}}$ has a Toeplitz structure just like $\mathbf{Q_{ss}}$. In that case, we suggest solving the Kriging equations via the FFT-based Preconditioned Conjugate Gradient (FFT-PCG) solver described below. Other techniques, with slightly lower performance and less adequate for combination with the superposition task, include look-ahead Schur algorithms and algorithms based on generalized displacement structures (e.g. Gallivan et al. 1996, Kailath and Sayed 1999, Barel et al. 2001) and are not discussed here.

The preconditioned conjugate gradient (PCG) method (Shewchuk 1994) iteratively solves a linear system of equations $\mathbf{Ax} = \mathbf{b}$. The PCG method converges in as many iteration steps as there are distinct eigenvalues of $\mathbf{A}$. If there exists a preconditioning matrix $\mathbf{V}$ which clusters the eigenvalues of the product $\mathbf{V^{-1}A}$ around unity, the algorithm will converge in only a few steps.

The advantage in this context is that PCG only requires evaluation of the product $\mathbf{Ax}$, which may be performed without explicitly storing or knowing all elements of $\mathbf{A}$. The same holds for applying the preconditioner. In the FFT-PCG algorithm, these steps are evaluated through convolution via FFT, i.e. in $\mathcal{O}(n \log_2 n)$ operations. Solving a circulant system is identical to convolution via FFT with only the Hadamard product replaced by

element-wise division (compare Good 1950, Rino 1970). If choosing a circulant matrix as preconditioner for the Toeplitz system, then applying the preconditioner is once again an $n \log_2 n$ operation, and only one column of the preconditioner needs to be stored.

Chan and Ng (1996) review and compare circulant preconditioners that have the same size as the original Toeplitz matrix and differ in the norm of $\mathbf{V}^{-1}\mathbf{A}$ they satisfy. For poor-conditioned Toeplitz systems, e.g., for Gaussian covariance matrices with high correlation lengths (e.g., Wesson and Pegram (2004)), the preconditioners themselves have a very poor condition, causing numerical noise to be amplified. Wesson and Pegram (2004) showed that singular value decomposition is very powerful in suppressing numerical artifacts when Kriging with such poorly conditioned systems. Trapp (1973) discussed generalized inverses of circulant matrices which could be used in this context, but we encountered poor convergence of generalized inverses with our iterative solver in preliminary studies. Along similar lines, Nowak (2005) proposed a regularization term, which is added to the diagonal of circulant preconditioners and evenly increases all eigenvalues (Appendix) instead of disregarding the almost-zero ones. We chose the Strang preconditioner (Strang 1986) on the embedded level, which is identical to the circulant matrix already in use for convolution via FFT, in conjunction with the regularization suggested by Nowak. However, if measurement error is included so that $\mathbf{R} \neq \mathbf{0}$, then $\mathbf{R}$ regularizes the entire system anyway and it is unlikely that additional regularization is required (compare Dietrich and Newsam 1989).

Figure 2 compares CPU times and storage requirements for standard Gaussian elimination (built-in solver for dense systems in MATLAB R2006b) versus the FFT-based PCG algorithm for the solution of Toeplitz systems of different sizes m. For a number of unknowns of $m \approx 300$, a break-even point of the two algorithms can be observed. Still below $m = 1,000$, FFT-based PCG is faster than the standard solver by more than an order of magnitude. At $m = 10,000$, the standard solver broke down due to memory overflow, whereas the FFT-based PCG worked up to $m = 1.6 \cdot 10^7$. The figure also demonstrates the superiority of the FFT-PCG over other standard solvers equipped with FFT-based algorithms for handling the Toeplitz matrix. Iterative Constrained Deconvolution (Pegram

2004, see also next section) is still faster than steepest descent (e.g., Press et al. 1992). The largest individual speedup is achieved by preconditioning, which requires more effort per iteration step but drastically reduces the number of iterations. The conjugation of gradients increases the effort per step once more, but the net effect of even fewer iteration steps prevails.

Extension of Toeplitz solvers to irregular grids

In order to solve Eq. 7 for irregularly scattered measurements, Pegram (2004) uses an algorithm called Iterative Constrained Deconvolution (ICD). The ICD algorithm can be identified as a steepest descent algorithm of the form $\boldsymbol{\xi}_{i+1} = \boldsymbol{\xi}_i + \alpha\left(\mathbf{Y} - \mathbf{Q_{yy}}\boldsymbol{\xi}_i\right)$, where α denotes a user-defined step size coefficient. For the matrix-vector product $\mathbf{Q_{yy}}\boldsymbol{\xi}_i$, he uses an injection/sampling combination

$$\mathbf{Q_{yy}}\boldsymbol{\xi}_i = \mathbf{H}\left[\mathbf{Q_{ss}}\left(\mathbf{H}^T\boldsymbol{\xi}_i\right)\right] \tag{18}$$

and evaluates the product with $\mathbf{Q_{ss}}$ using convolution via FFT. To be more precise, the sampling (multiplication with $\mathbf{H}$) is never actually performed during the algorithm. Instead, the final value within the square brackets is used for $\mathbf{Q_{sy}}\boldsymbol{\xi}$ in Eq. 6, thus avoiding one additional superposition.

In order to extend the FFT-based PCG to irregular sampling grids, we seized Pegram's suggestion to inject the irregularly scattered measurements into a finer regular grid, as also done in Eq. 18. The corresponding injection/sampling procedure $\mathbf{Q_{yy}} = \mathbf{H}\mathbf{Q_{ss}}\mathbf{H}^T$, where $\mathbf{Q_{ss}}$ is a larger Toeplitz matrix, is formally identical to the embedding/extraction procedure $\mathbf{Q_{ss}} = \mathbf{M}^T\mathbf{Q_{ss,}}_e\mathbf{M}$ to embed a Toeplitz matrix in a larger circulant matrix, and both procedures can be combined into one:

$$\begin{aligned}
\mathbf{Q_{yy}} &= \mathbf{H}\mathbf{Q_{ss}}\mathbf{H}^T \\
\mathbf{Q_{ss}} &= \mathbf{M}^T\mathbf{Q_{ss,}}_e\mathbf{M} \\
\rightarrow \mathbf{Q_{yy}} &= \left(\mathbf{H}\mathbf{M}^T\right)\mathbf{Q_{ss,}}_e\left(\mathbf{M}\mathbf{H}^T\right)
\end{aligned} \tag{19}$$

All other particulars are identical to the FFT-based PCG algorithm for Toeplitz matrices (regular measurement grids). Placed within the PCG framework, the idea of sampling/injection is much more efficient than within the framework of steepest descent as used in the ICD algorithm. The resulting FFT-based PCG solver for almost-Toeplitz matrices (irregularly scattered measurements) is described in Appendix.

Figure 3 compares CPU times and storage requirements for FFT-based solvers for nearly-Toeplitz systems with those of standard Gaussian elimination for different problem size m and sizes of the finer regular grid n. The finer grid introduces a substantial overhead if a high spatial accuracy of measurement locations on the finer grid is desired. Therefore, the standard solver is faster than the FFT-PCG in the majority of cases. However, the reduced storage requirements allow FFT-based algorithms to solve much larger systems of equations. On our reference computer, the limiting size of the underlying regular grid was $n = 1.6 \cdot 10^7$. The FFT-PCG solver proposed in this study outperforms the ICD algorithm by a factor of roughly ten.

APPLICATION OF FFT-BASED ALGORITHMS TO KRIGING

Kriging Estimator

In order to apply the efficient solvers discussed above for solving the Kriging system of equations (Eq. 7), we exploit the fact that the coefficient matrix (also called the Kriging matrix) consists of an $m \times m$ sub-matrix $\mathbf{Q_{yy}}$ with exploitable structure and structural perturbations of rank $p \ll m$. First, we introduce three auxiliary variables which will be re-used several times in the following procedures:

$$\mathbf{x} = \mathbf{HX} \tag{20}$$

$$\mathbf{y} = \mathbf{Q_{yy}}^{-1}\mathbf{Y} \tag{21}$$

$$\mathbf{z} = \mathbf{Q_{yy}}^{-1}\mathbf{HX} = \mathbf{Q_{yy}}^{-1}\mathbf{x} \tag{22}$$

Then, we partition the inverse of the Kriging matrix in 7 as follows (Kitanidis 1996, Nowak and Cirpka 2004):

$$\begin{bmatrix} \mathbf{Q_{yy}} & \mathbf{HX} \\ \mathbf{X}^T\mathbf{H}^T & -\mathbf{Q}_{\boldsymbol{\beta\beta}}^{-1} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{P_{yy}} & \mathbf{P}_{\mathbf{y}\boldsymbol{\beta}} \\ \mathbf{P}_{\boldsymbol{\beta}\mathbf{y}} & \mathbf{P}_{\boldsymbol{\beta\beta}} \end{bmatrix} \tag{23}$$

Now, we express the sub-matrices $\mathbf{P}$ in Eq. (23) according to Schweppe (1973) and immediately simplify using our auxiliary quantities:

$$\mathbf{P}_{\boldsymbol{\beta\beta}} = -\left(\mathbf{x}^T\mathbf{z} + \mathbf{Q}_{\boldsymbol{\beta\beta}}^{-1}\right)^{-1} \tag{24}$$

$$\mathbf{P}_{\boldsymbol{\beta}\mathbf{y}} = \mathbf{P}_{\mathbf{y}\boldsymbol{\beta}}{}^T = -\mathbf{P}_{\boldsymbol{\beta\beta}}\mathbf{z}^T \tag{25}$$

$$\mathbf{P_{yy}} = \mathbf{Q_{yy}}{}^{-1} + \mathbf{z}\mathbf{P}_{\boldsymbol{\beta\beta}}\mathbf{z}^T \tag{26}$$

where Eq. 24 is also known as the Schur complement. Using these sub-matrices yields a partitioned form of the coefficient vector (compare Nowak and Cirpka 2004):

$$\begin{aligned} \boldsymbol{\xi} &= \mathbf{y} - \mathbf{z}\widehat{\boldsymbol{\beta}} \\ \widehat{\boldsymbol{\beta}} &= -\mathbf{P}_{\boldsymbol{\beta\beta}}\left(\mathbf{z}^T\mathbf{Y} + \mathbf{Q}_{\boldsymbol{\beta\beta}}^{-1}\boldsymbol{\beta}*\right) \end{aligned} \tag{27}$$

Altogether, the entire estimator requires to:

1. Compute the auxiliary quantities $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$ according to Eqs. (20) to (22), using the most appropriate solver (e.g. the FFT-based PCG),
2. Evaluate the partitioned solution vector of the Kriging system according to Eq. (27),
3. Evaluate the estimate according to Eq. (6), using superposition via FFT (Eq. 17) for $\mathbf{Q_{sy}}\boldsymbol{\xi}$ and simple matrix-vector multiplication for $\mathbf{X}\widehat{\boldsymbol{\beta}}$

Step one requires p sampling processes to obtain $\mathbf{x}$, and $1+p$ solutions of a (nearly) Toeplitz system to obtain $\mathbf{y}$ and $\mathbf{z}$, with an asymptotic cost estimate of $\mathcal{O}\left((1+p)\, m \log_2 m\right)$ (when $\mathbf{Q_{ss}}$ is second-order stationary and the measurements $\mathbf{y}$ fall onto a regular grid so that the FFT-PCG for regular grids can be applied). Step two involves only smaller operations of

$\mathcal{O}(mp)$ and $\mathcal{O}(p^2)$ to treat the rank p perturbations in the structure of the Kriging matrix. Step three is an $\mathcal{O}(np + n\log_2 n)$ operation. This yields a total of $\mathcal{O}(n\log_2 n + m\log_2 m)$ for the entire estimation with storage requirements of only $\mathcal{O}(n)$, given that $n \gg m \gg p$.

Estimation Variance

To obtain an efficient procedure for evaluating the estimation variance, we re-combine some terms in Eq. (8):

$$\mathbf{Q_{ss|y}} = \mathbf{Q_{ss}} - \underbrace{\begin{bmatrix} \mathbf{Q_{ys}} \\ \mathbf{X}^T \end{bmatrix}^T \begin{bmatrix} \mathbf{Q_{yy}} & \mathbf{HX} \\ (\mathbf{HX})^T & -\mathbf{Q_{\beta\beta}^{-1}} \end{bmatrix}^{-1}}_{\mathbf{S}} \begin{bmatrix} \mathbf{Q_{ys}} \\ \mathbf{X}^T \end{bmatrix} \tag{28}$$

$$\mathbf{S} = \begin{bmatrix} \mathbf{Q_{ys}} \\ \mathbf{X}^T \end{bmatrix}^T \begin{bmatrix} \mathbf{Q_{yy}} & \mathbf{HX} \\ (\mathbf{HX})^T & -\mathbf{Q_{\beta\beta}^{-1}} \end{bmatrix}^{-1} \mathbf{I}_{(m+p)} \tag{29}$$

Here, we appended the $(m+p)\times(m+p)$ identity matrix $\mathbf{I}$ to $\mathbf{S}$ without actually changing the equation. However, it clarifies that the $(m+p)$ columns of $\mathbf{S}$ are unit estimators $\mathbf{s}_i$ that arise from using the $(m+p)$ unit vectors $\mathbf{e}_i$ in $\mathbf{I}_{(m+p)}$ as data vectors for Kriging in Eq. (7).

For the estimation variance $\hat{\boldsymbol{\sigma}}^2$, we only need the diagonal of $\mathbf{Q_{ss|y}}$. The diagonal of a matrix product or of dyadic matrices (i.e., matrices defined by dyadic products) with low rank can be directly and efficiently evaluated using

$$diag\left(\mathbf{A}_{(n\times m)}\mathbf{B}^T_{(n\times m)}\right) = \sum_{i=1}^{m}\left(\mathbf{a}_i \circ \mathbf{b}_i\right) \tag{30}$$

where $\mathbf{a}_i$ and $\mathbf{b}_i$ are the i-th columns of $\mathbf{A}$ and $\mathbf{B}$, respectively. Then, Eq. (28) simplifies to:

$$\hat{\boldsymbol{\sigma}}^2 = diag\left(\mathbf{Q_{ss|y}}\right) = \sigma^2 - \sum_{i=1}^{m+p} \underbrace{\mathbf{s}_i \circ \begin{bmatrix} \mathbf{Q_{sy}} & \mathbf{X} \end{bmatrix}_i}_{\sigma^2_{kriging}} \tag{31}$$

To efficiently evaluate this expression:

1. Initialize $\hat{\boldsymbol{\sigma}}^2 = \sigma^2$,
2. Evaluate the unit estimator $\mathbf{s}_i$ for the $(m+p) \times 1$ unit vector $\mathbf{e}_i$ as data vector for Kriging, using FFT-based algorithms and Eqs. (20) to (27) as described in the previous section,
3. Perform the Hadamard product of the unit estimator with the i-th column of $\begin{bmatrix} \mathbf{Q_{sy}} & \mathbf{X} \end{bmatrix}$ and subtract the result from $\hat{\boldsymbol{\sigma}}^2$ and
4. Repeat steps 2 and 3 for $i = 1 \dots m+p$.

In total, this makes evaluating $\hat{\boldsymbol{\sigma}}^2$ an effort of $(m+p)$ estimations, $(m+p)$ Hadamard products and summation of $(m+p)$ vectors, each sized $n \times 1$, resulting in an asymptotic complexity of $\mathcal{O}\left(mn \log_2 n + m^2 \log_2 m\right)$.

Efficient Approximations to the Estimation Variance

For the sake of the following analysis, we revert to the partitioned form of the Kriging matrix. Starting from Eq. (8), inserting Eqs. 24 to 26 and applying Eq. 30 yields after some rearrangement:

$$\begin{aligned} \sigma^2_{kriging} \;=\; & \sum_{i=1}^{m} \left[\mathbf{Q_{sy}} {\mathbf{Q_{yy}}}^{-1}\right]_i \circ \mathbf{Q_{sy},}_i + \sum_{i=1}^{p} \left[\mathbf{Q_{sy}} \mathbf{z} \mathbf{P}_{\boldsymbol{\beta\beta}}\right]_i \circ \left[\mathbf{Q_{sy}} \mathbf{z}\right]_i \\ & -2 \sum_{i=1}^{p} \left[\mathbf{Q_{sy}} \mathbf{z} \mathbf{P}_{\boldsymbol{\beta\beta}}\right]_i \circ \mathbf{X}_i + \sum_{i=1}^{p} \left[\mathbf{X} \mathbf{P}_{\boldsymbol{\beta\beta}}\right]_i \circ \mathbf{X}_i \end{aligned} \tag{32}$$

The first term is the one required for Kriging with known mean, i.e. with a number of base functions $p = 0$, and represents m unit estimators similar to those defined in Eq. 29. Hence, it has well-known properties. The approximations in the following sections will focus on the asymptotic behavior of this first term under specific conditions. With no further simplifications, this term requires full inversion of $\mathbf{Q_{yy}}$, m superpositions via FFT to evaluate $\mathbf{Q_{sy}} {\mathbf{Q_{yy}}}^{-1}$ and then m Hadamard vector-vector products, followed by summation over index i.

All other terms relate to the uncertainty in estimating the trend coefficients. We will not simplify them any further, since their computational complexity is negligible over that of the first term, given that $m \gg p$: We first evaluate p superpositions of $\mathcal{O}(n \log_2 n)$ each to obtain the individual columns of the $n \times p$ matrix $\boldsymbol{\zeta} = \mathbf{Q_{sy}} \mathbf{z}$ via FFT. The remaining steps are $\mathcal{O}(np^2)$ for the multiplication with $\mathbf{P}_{\boldsymbol{\beta\beta}}$ and the $3p$ Hadamard products.

Single-Point Approximations

In the single-point approximations discussed here, correlation between measurements is neglected. They have been suggested by Pegram (2004) for zero measurement error and the specific cases of known/unknown mean. These approximations reduce the computational complexity of the estimation variance by one order in m.

1. For vanishing correlation among the measurement points on both irregular and regular grids, $\mathbf{Q_{yy}}$ approaches a scalar matrix $\mathbf{Q_{yy}} \approx (\sigma^2 + \sigma_{err}^2)\,\mathbf{I}$ with σ_{err}^2 being the variance of measurement error. Then, the first term can be approximated by:

$$\sum_{i=1}^{m} \left[\mathbf{Q_{sy}} \mathbf{Q_{yy}}^{-1}\right]_i \circ \mathbf{Q_{sy},}_i \;\; \approx \tfrac{1}{\sigma^2+\sigma_{err}^2} \;\; \sum_{i=1}^{m} \mathbf{Q_{sy},}_i \circ \mathbf{Q_{sy},}_i \tag{33}$$

2. For the case of known-mean Kriging, the estimation variance is equal to the value of the measurement error variance σ_{err}^2 at measurement locations. If the correlation among measurements is too large for the above simplification, the latter condition may still be enforced by requiring that the first term equals $\sigma^2 - \sigma_{err}^2$ at the locations of measurements. This is achieved by solving the subsidiary Kriging problem:

$$\sum_{i=1}^{m} \left[\mathbf{Q_{sy}} \mathbf{Q_{yy}}^{-1}\right]_i \circ \mathbf{Q_{sy},}_i \;\; \approx \;\; \mathbf{Q}_{\mathbf{sy}}^{*} \left(\mathbf{Q}_{\mathbf{yy}}^{*}\right)^{-1} \mathbf{u}_m \left(\sigma^2 - \sigma_{err}^2\right) \tag{34}$$

where $\mathbf{u}_m$ is a $m \times 1$ vector of ones used as data vector in the subsidiary problem, $\mathbf{Q_{ss}}^{*} = \sigma^{-2} \mathbf{Q_{ss}} \circ \mathbf{Q_{ss}}$, $\mathbf{Q}_{\mathbf{sy}}^{*} = \mathbf{Q_{ss}}^{*} \mathbf{H}^T$, and $\mathbf{Q_{yy}}^{*} = \mathbf{H} \mathbf{Q_{ss}}^{*} \mathbf{H}^T$.

Infinite Regular Grid Approximations

If the grid of measurements is regular and equispaced and its extent is much larger than the

range of correlation, the number of mutually correlated measurements differs only within a thin boundary area. Besides that area, the inner section of the measurement behaves statistically stationary. If the application permits, one may neglect these boundary effects and assume that all measurement points have the same correlation to the same number of neighbors. Then, the contribution of one single representative measurement point in the center of the domain towards the estimation variance may be used to approximate that of all others by shifting:

$$\mathbf{Q_{sy}}\mathbf{Q_{yy}}^{-1} \;\approx\; \mathrm{shift}_m\left[\mathbf{Q_{sy}}\boldsymbol{\xi}_c\right] \tag{35}$$

The term within square brackets is a $n \times 1$ zero-mean Representative Unit Estimator (RUE) for a unit measurement in the center of the domain, and $\mathrm{shift}_m[\cdot]$ denotes an $n \times m$ matrix obtained from concatenating the RUE m times, each time shifted to the respective next measurement location. To allow this shifting operation, the RUE needs to be larger than the actual domain. This is achieved by performing the superposition $\mathbf{Q_{sy}}\boldsymbol{\xi}_c$ for a virtual larger domain, using the embedded version of the covariance function.

Since measurements in the interior convey less independent information than those at the boundaries, this approximation yields a conservative upper bound of the estimation variance. It is exact for the case of an infinite (or periodic) grid of measurements. The remaining complexity is again reduced by one order in m.

Hybrid Regular Grid Approximation

In the infinite regular grid approximation, the largest errors occur along the boundaries of the measurement grid. The corresponding unit estimators may be evaluated separately in their exact form, whereas the bulk inner part of the unit estimators are approximated by shifting the RUE. A user-defined break criterion for the observed difference between RUE and exact unit estimator can be used to determine the boundary zone in which to use the exact version.

Infinite Regular Grid Average

Another option to simplify the estimation variance is to start with a conditional covariance made stationary by averaging over all starting points of separation vectors with same length, i.e. by averaging each diagonal of $\mathbf{Q_{ss|y}}$. A semi-analytical solution for the averaged conditional covariance has been developed by Cirpka and Nowak (2003). We will translate the formalism into the notation and context used here for the sake of completeness. Due to averaging, the diagonals $diag\,(\cdot)$ in Eqs. (30) and (31) become traces $Tr\,(\cdot)$ multiplied by a factor of $1/n$. Since the trace is invariant with respect to cyclic permutations, the averaged equivalent of Eq. (32) is:

$$\begin{aligned}\overline{\sigma^2_{kriging}} =& \frac{1}{n} Tr\left(\mathbf{Q_{yy}}^{-1}\mathbf{Q_{ys}}\mathbf{Q_{sy}}\right) + \frac{1}{n} Tr\left(\mathbf{P}_{\boldsymbol{\beta\beta}}\mathbf{z}^T\mathbf{Q_{ys}}\mathbf{Q_{sy}}\mathbf{z}\right) \\ & - \frac{2}{n} Tr\left(\mathbf{P}_{\boldsymbol{\beta\beta}}\mathbf{X}^T\mathbf{Q_{sy}}\mathbf{z}\right) + \frac{1}{n} Tr\left(\mathbf{P}_{\boldsymbol{\beta\beta}}\mathbf{X}^T\mathbf{X}\right) \end{aligned} \tag{36}$$

In this form, all terms but the first have boiled down to traces of $p \times p$ matrices. If we again evaluate $\boldsymbol{\zeta} = \mathbf{Q_{sy}}\mathbf{z}$ via FFT, these small matrices are obtained from $3p$ scalar products and are computationally cheap.

For regular measurement grids, the first term is the inverse of an $m \times m$ Toeplitz matrix multiplied with another $m \times m$ Toeplitz matrix $\mathbf{Q_{ys}}\mathbf{Q_{sy}} = \mathbf{H}\mathbf{Q_{ss}}\mathbf{Q_{ss}}\mathbf{H}^T$. For small numbers of measurements, it may be evaluated exactly in a manner similar to the unit estimator technique described above, but the procedure may quickly explode in computational effort for larger m. If the measurement grid is sufficiently large or periodic, we may approximate both matrices by circulant counterparts, and evaluate first column of the circulant resulting from the martrix product by convolution via FFT of the individual first columns. The trace of the resulting circulant is quite trivially given by m times the first element (e.g., Davis 1979).

Generation of unconditional and conditional random fields

The fast and exact generation of random fields via FFT has been described by Dykaar

and Kitanidis (1992) and by Dietrich and Newsam (1993). The generation of statistically stationary random fields on regular grids is based on the symmetric decomposition of $\mathbf{Q_{ss}}$ achieved by taking the element-wise square root of is eigenvalues obtained from $\mathcal{F}[\mathbf{q_{ss},}_e]$, followed by convolution via FFT with a white-noise vector. The overall computational effort is as low as $\mathcal{O}(n \log_2 n)$ at storage requirements of, once again, only $\mathcal{O}(n)$. Conditioning to direct or linearly dependent measurements has been dealt with by Dietrich and Newsam (1996). For the non-linear case, the same procedure is suggested by Kitanidis (1995). They correct unconditional realizations to conditional ones by subtracting a Kriging estimator with adequately chosen measurements. Without going into further details, we point out that the FFT-based algorithms for Kriging described in this study can also be applied in this correction step, keeping the asymptotic cost estimate down at $\mathcal{O}(mn \log_2 n + m^2 \log_2 m)$ and memory consumption at $\mathcal{O}(n)$.

PERFORMANCE TESTS

In this section, the performance of the described FFT-based methods is demonstrated in comparison with standard methods. Our performance analysis was carried out on a contemporary desktop computer (i386, 2.8GHz Intel Xeon dual-core, 2GB RAM, Suse Linux 9.2) purchased in 2004. All methods were implemented in MATLAB (Release 2006b). As conventional solver for dense systems, the built-in Gaussian elimination in MATLAB was used. MATLAB also includes the Basic Linear Algebra Package and an implementation of the FFTW (Frigo and Johnson 1998). In order to keep the FFTW algorithm efficient for arbitrary domain sizes, we implemented a small algorithm that chooses embedding sizes with prime factors of 2, 3, 5 and 7 only. Required relative error norm for all iterative solvers was set to 10^{-10}.

In all performance tests, we used random measurement data and varied both the number n of estimation points and the number m of measurements. All tests assumed an uncertain but spatially constant mean value, so that the number p of trend coefficients is one and $\mathbf{X}$ is a $n \times 1$ vector of ones. The individual test series are composed of $n = 2^k$, $k = 2 \ldots 24$ estimation points and ratios between n and the number m of estimation points given by

$n/m = 2^\ell$, $\ell = 2 \ldots 14$. All computations above $10^5 s$ (approx. 1 day) were stopped and their CPU times estimated by extrapolation of fitted complexity models.

Kriging with conventional solver

Standard Kriging on a regular estimation grid and with irregularly scattered measurements serves as our base-case for comparison. With standard Kriging, we refer to Gaussian elimination for solving the Kriging system, conventional superposition by successive addition of the shifted covariance kernel, and evaluation of the exact estimation variance

The results are displayed in Figure 4 (solid lines). The auto-covariance matrix of the unknowns $\mathbf{s}$ is represented by its first column only in our conventional superposition algorithm. Hence, mainly the size of the Kriging matrix, i.e. the number m of measurements squared, is the limiting factor to the problem size. Had we chosen a brute-force matrix product approach to superposition and had we explicitly stored the $n \times m$ matrix $\mathbf{Q_{sy}}$, the limitation would have been even more severe. Due to the restriction in size, CPU time never rose above one day. When using the one-point approximation to simplify the estimation variance (not shown here), the computational effort reduces by a factor of roughly m.

The speedup of superposition via FFT versus conventional superposition can be read from the dashed lines in Figure 4. Since the overall complexity is dominated by the solution of the system, the order of the asymptotic cost estimate does not change for high measurement numbers, but speedup factors of up to fifty occur. The memory-related restrictions of measurement numbers still holds, and the storage requirements of both versions coincide.

Kriging with FFT-based solvers

Now, we demonstrate the effect of using the FFT-based PCG solver instead of Gaussian elimination, combined with superposition via FFT (Figure 5). The FFT-based PCG solver is slightly slower for unfavorable ratios n/m (compare Figure 3), but radically cancels the dependence of storage requirements on the number of measurements. As a consequence,

only the grid of estimation limits the admissible problem size. For small numbers of measurements, it is more efficient to revert to conventional Gaussian elimination (dashed lines in Figure 4).

The CPU time for evaluating the estimation variance is $\mathcal{O}(m)$ larger than that of one Kriging estimate. Figure 5 also shows how approximations to the estimation variance can drastically reduce the overall computational effort by approximately this factor of m. For large numbers of measurements (small n/m), the observed advantage in CPU time rises up to five orders of magnitude. When evaluating only the estimator and omitting the estimation variance (not shown here), CPU times would show the same shape of dependence on n and n/m as the dashed lines, reduced by a factor of approximately two.

The greatest advantage can be made if the measurements are on a regular grid. For that case, the FFT-based PCG solver outruns Gaussian elimination for any problem size and permits vast numbers of measurements at even lower computational expense. The CPU times are smaller by about one order of magnitude over a large range of problem sizes (Figure 6). Especially for large problems, the infinite-grid approximation of the estimation variance is likely to be sufficiently accurate for many purposes. The associated computational speedup is again about ten compared to the irregular-grid case, and again about 5 orders of magnitude compared to the exact evaluation of the estimation variance. Entirely omitting the estimation variance (not shown here), as in the previous case, results in a reduction of CPU times down to roughly one half of the dashed lines.

SUMMARY AND CONCLUSIONS

In this study we compiled a toolbox of existing and newly extended FFT-based methods to drastically speed up Kriging for large estimation problems. These methods include FFT-based convolution, FFT-based superposition and an FFT-based PCG solver. All methods apply to estimation on regular and equispaced grids. The measurements may either lie on regular and equispaced grids or be irregularly scattered. For irregularly scattered measurements, we introduced an extension of the existing FFT-based PCG method which can be understood as an extension of Pegram's (2004) method, where

scattered measurements are assigned to nodes of a (finer) regular grid. The density of the regular grid is dictated by the required accuracy in discretizing the irregular grid, depending on the specific application. We showed how to apply these methods to the generalized case of Kriging with uncertain mean and trends, including measurement error. The efficiency of all methods was demonstrated in a series of performance tests on an ordinary desktop computer with all algorithms implemented in MATLAB.

The first eye-catching advantage of all FFT-based methods is that only the covariance function for one point, i.e. the first column of the respective auto-covariance matrices, needs to be stored. This makes the storage requirements shrink from $\mathcal{O}(n^2)$ and $\mathcal{O}(m^2)$ for whole covariance matrices to only $\mathcal{O}(n)$ and $\mathcal{O}(m)$ for one column each (where n is the number of estimation points and m is the number of measurements). This also reduces the time needed to set up the matrices themselves, which is not negligible since evaluating effective separation distances and covariance functions (including exponentials) has substantial computational costs.

FFT-based superposition has computational costs in the order of $n \log_2 n$ instead of mn, which turned out to be highly favorable even for very small data sets on our reference computer. For regular and equispaced measurement grids, the FFT-based PCG solver with its complexity of $\mathcal{O}(m \log_2 m)$ outruns standard solvers for dense systems by far. For irregularly scattered measurements, the extended FFT-based PCG solver is less efficient than the standard solver for low and medium numbers of measurements. The break-even point is at roughly $m = 1000$ measurements, depending on the resolution of the finer regular grid. The great advantage, however, is that the FFT-based solver does not require to store the $m \times m$ coefficient matrix, so it is applicable for arbitrarily high m. Any solver that requires to explicitly store the dense $m \times m$ Kriging broke down due to memory restrictions at a maximum of $m = 10,000$ on our reference computer. Combining these methods, the Kriging estimator can be evaluated up to several millions of estimation points in no more than seconds up to a few minutes on contemporary desktop computers.

Evaluation of the estimation variance is computationally much more demanding. For the evaluation of the exact estimation variance, an equivalent of m estimation procedures

has to be performed, which may turn out to be strictly inhibiting for larger data sets of measurements, in spite of the speedup achieved by FFT-based methods. We alleviated this situation by reviewing, extending and proposing several fast approximations which are asymptotically exact in specific cases. These cases include negligible correlation among the measurements and very large regular grids of measurements. The approximations offer an additional speedup of up to five orders of magnitude for large m, so that the largest admissible problem could be completed in less than one day.

In summary, the methods reviewed, discussed and introduced in this paper greatly reduce the computational effort and storage requirements of Kriging, allowing to handle several millions of estimation points and thousands of measurements on ordinary desktop computers.

OUTLOOK

For the near future, we expect further advances in the development of non-uniform FFT algorithms (NUFFT), also called the generalized FFT (e.g., Liu and Ngyuen 1998, Duijndam and Schonewille 1999, Fourmont 2003, Fessler and Sutton 2003, Greengard and Lee 2004). The Kriging context requires NUFFT algorithms for frequency-space data on a regular grid with real-space data on an irregular grid, that work efficiently in both transform directions. We assume that highly efficient algorithms for these requirements will be readily available within a few years. Such algorithms will help to further increase the efficiency of preconditioning the PCG solver for irregularly scattered measurements.

ACKNOLEDGMENTS

We would like to thank Prof. Pegram at the University of KwaZulu-Natal, South Africa, for his helpful comments on the manuscript. This study was fundet in Parts by Deutsche Forschungsgemeinschaft (DFG) under the grants Ne 824/2-2 and Ne 824/3.

APPENDIX

FFT-based PCG for irregular and regular grids

The Conjugate Gradients Method is attributed to Hestenes and Stiefel (1952). The following is the preconditioned version taken from Shewchuk (1994), combined with convolution via FFT (e.g. van Loan 1992) and the embedded regularized preconditioner by Nowak (2005). For an extensive review of circulant preconditioners used in PCG algorithms, see Chan and Ng (1996).

The Fourier transform of the first column of a circulant matrix yields the eigenvalues, thanks to the diagonalization theorem (e.g., Trapp 1973, Barnett 1990 pp. 350-354). If $\mathbf{C}$ is a (level-d) real circulant matrix with first column $\mathbf{c}$ and $\tilde{\mathbf{c}} = \mathcal{F}_d(\mathbf{c})$ is the (d-dimensional) Fourier transform of the first column, then the condition c of $\mathbf{C}$ is the ratio of the largest value c_{max} and the smallest value c_{min} in $\tilde{\mathbf{c}}$. The regularized preconditioner by Nowak (2005) installs a maximum condition c^*(e.g. 10^5) of the preconditioner through a diagonal regularization $\mathbf{C}* = \mathbf{C} + \varepsilon^* \mathbf{I}$, where ε is chosen according to

$$\varepsilon* = \frac{c_{max} - c_{min} c*}{c* - 1} \ . \tag{37}$$

Automatically, the Fourier transform $\tilde{\mathbf{c}}^* = \mathcal{F}_d(\mathbf{c}^*)$ of the first column of $\mathbf{C}^*$ is given by $\tilde{\mathbf{c}}^* = \tilde{\mathbf{c}} + \varepsilon^*$.

In the following, $\circ$ denotes the element-wise (Hadamard) product and $\div$ denotes element-wise division.

Algorithm 1 *(Preconditioned Conjugate Gradients with circulant preconditioning for nearly-Toeplitz system): The linear system* $\mathbf{Ax} = \mathbf{b}$ *is to be solved for a real symmetric positive-definite* $m \times m$ *matrix* $\mathbf{A} = \mathbf{HTH}^T$*, where* $\mathbf{H}$ *is a sampling matrix as defined in Eq. 2 and* $\mathbf{T}$ *is a (level-d) symmetric positive-definite Toeplitz matrix sized* $n \times n$*.* $\mathbf{T}$ *has a symmetric positive-definite embedding circulant matrix* $\mathbf{C}$ *with* $\mathbf{T} = \mathbf{M}^T \mathbf{C} \mathbf{M}$*, where* $\mathbf{M}$ *is a mapping matrix as defined in Eq. 9.* $\mathbf{c}$ *is the first column of* $\mathbf{C}$ *with (d-dimensional) Fourier transform* $\tilde{\mathbf{c}} = \mathcal{F}_d(\mathbf{c})$*. An initial guess*

$\mathbf{x}_0$, *an error tolerance* $\varepsilon < 1$ *and a maximum allowable condition* c^* *are provided. Initialize the algorithm with counter* $k = 1$, *error vector* $\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{x}_0$, *the preconditioned conjugate gradient* $\mathbf{d} \approx \mathbf{A}^{-1}\mathbf{r}$, *the residual* $\delta_1 = \mathbf{r}^T\mathbf{d}$, *the initial residual* $\delta_0 = \delta_1$ *and evaluate* ε^* *according to Eq. (37). Then,*

1. *update the trial solution* $\mathbf{x}$ *using:*

$$\begin{aligned} \mathbf{q} &= \mathbf{H}\mathbf{M}^T\mathcal{F}_d^{-1}\left[\mathcal{F}_d\left[\mathbf{M}\mathbf{H}^T\mathbf{d}\right] \circ \tilde{\mathbf{c}}\right] \quad (= \mathbf{A}\mathbf{d}) \\ \alpha &= \frac{\delta_k}{\mathbf{d}^T\mathbf{q}} \\ \mathbf{x} &= \mathbf{x} + \alpha\mathbf{d} \end{aligned}$$

2. *update the error vector and residual:*

$$\begin{aligned} \mathbf{r} &= \mathbf{r} - \alpha\mathbf{q} \\ \mathbf{s} &= \mathbf{H}\mathbf{M}^T\mathcal{F}_d^{-1}\left[\mathcal{F}_d\left[\mathbf{M}\mathbf{H}^T\mathbf{r}\right] \div (\tilde{\mathbf{c}} + \varepsilon^*)\right] \quad \left(\approx \mathbf{A}^{-1}\mathbf{r}\right) \\ \delta_{k+1} &= \mathbf{r}^T\mathbf{s} \end{aligned}$$

3. *Update the preconditioned conjugate gradient*

$$\mathbf{d} = \mathbf{s} + \frac{\delta_{k+1}}{\delta_k}\mathbf{d}$$

4. *Increase* k *by one and repeat until* $k > k_{max}$ *or* $\delta_{k+1} < \varepsilon^2\delta_0$.

The variables $\mathbf{q}$, α and $\mathbf{s}$ are auxiliaries to reduce the computational costs. All products with $\mathbf{M}$ and $\mathbf{H}$ are evaluated by simple embedding/extraction and injection/sampling or using sparse representations of the matrix $\mathbf{N} = \mathbf{M}\mathbf{H}^T$. The PCG algorithm requires only one matrix-vector product per iteration step evaluated via FFT, which has an asymptotic cost estimate of $\mathcal{O}(n \log_2 n)$ and another operation of $\mathcal{O}(n \log_2 n)$ to apply the preconditioner. The corresponding steps in the initialization are evaluated accordingly, resulting in an overall complexity of $\mathcal{O}(n \log_2 n)$. Here, the finer regular grid with n nodes is not

necessarily as large (as fine) as the grid of estimation, subject to the desired accuracy of discretizing the locations of the measurements.

In case the measurements themselves lie on a regular grid, matrix $\mathbf{A} = \mathbf{T}$ in the above algorithm is a Toeplitz matrix itself, and $\mathbf{H} = \mathbf{H}^T = \mathbf{I}$ may be omitted in the entire algorithm. In that case, the computational complexity drops to $\mathcal{O}(m \log_2 m)$.

References

Barel, M. V., Heinig, G., and Kravanja, P. (2001). A stabilized superfast solver for nonsymmetric Toeplitz systems. *SIAM J. Matrix Anal. A.*, 23(2):494–510.

Barnett, S. (1990). *Matrices Methods and Applications*. Oxford Applied Mathematics and Computing Science Series. Clarendon Press, Oxford.

Chan, R. H. and Ng, M. K. (1996). Conjugate gradient methods for Toeplitz systems. *SIAM Review*, 38(3):427–482.

Cirpka, O. A. and Nowak, W. (2003). Dispersion on kriged hydraulic conductivity fields. *Water Resour. Res.*, 39(2):doi:10.1029/2001WR000598.

Cirpka, O. A. and Nowak, W. (2004). First-order variance of travel time in non-stationary formations. *Water Resour. Res.*, 40:doi:10.1029/2003WR002851.

Cooley, J. W. and Tukey, J. W. (1965). An algorithm for the machine calculation of complex Fourier series. *Math. Comput.*, 19:297–301.

Davis, P. J. (1979). *Circulant Matrices*. Pure and Applied Mathematics. John Wiley and Sons, New York.

Dietrich, C. R. and Newsam, G. N. (1989). A stability analysis of the geostatistical approach to aquifer transmissivity identification. *Stoch. Hydrol. Hydraul.*, 3:293–316.

Dietrich, C. R. and Newsam, G. N. (1993). A fast and exact method for multidimensional Gaussian stochastic simulations. *Water Resour. Res.*, 29(8):2861–2869.

Dietrich, C. R. and Newsam, G. N. (1996). A fast and exact method for multidimensional Gaussian stochastic simulations: Extension to realizations conditioned on direct and indirect measurements. *Water Resour. Res.*, 32(6):1643–1652.

Dietrich, C. R. and Newsam, G. N. (1997). Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix. *SIAM J. ScI. Comput.*, 18(4):1088–1107.

Duijndam, A. J. W. and Schonewille, M. A. (1999). Nonuniform fast Fourier transform. *Geophys.*, 64(2):539–551.

Dykaar, B. B. and Kitanidis, P. K. (1992). Determination of the effective hydraulic conductivity for heterogeneous porous media using a numerical spectral approach. 1. Method. *Water Resour. Res.*, 28(4):1155–1166.

Fessler, J. A. and Sutton, B. P. (2003). Nonuniform fast Fourier transform using min-max interpolation. *IEEE Transactions on Signal Processing*, 51(2):560–574.

Fourmont, K. (2003). Non-equispaced fast Fourier transforms with applications to tomography. *J. Fourier Anal. Appl.*, 9(5):431–450.

Frigo, M. and Johnson, S. G. (1998). FFTW: An adaptive software architecture for the FFT. In *Proc. ICASSP*, volume 3, pages 1381–1384, IEEE, Seattle, WA. http://www.fftw.org.

Gallivan, K., Thirumalai, S., Dooren, P. V., and Vermaut, V. (1996). High performance algorithms for Toeplitz and block Toeplitz matrices. *Linear Algebra Appl.*, 241-243(1-3):343–88.

Golub, G. H. and van Loan, C. F. (1996). *Matrix Computations*. Jon Hopkins University Press, Baltimore, Md, third edition.

Good, I. J. (1950). On the inversion of circulant matrices. *Biometrika*, 37:185–186.

Greengard, L. and Lee, J.-Y. (2004). Accelerating the nonuniform fast Fourier transform. *SIAM Review*, 46(3):443–454.

Hestenes, M. R. and Stiefel, E. (1952). Methods of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Stand.*, 49:409–436.

Kailath, T. and Sayed, A. H. (1995). Displacement structure: Theory and applications. *SIAM Review*, 37(3):297–386.

Kailath, T. and Sayed, A. H. (1999). *Fast Reliable Algorithms of Matrices with Structure*. SIAM, Philadelphia, PA.

Kitanidis, P. K. (1993). Generalized covariance functions in estimation. *Math. Geol.*, 25(5):525–540.

Kitanidis, P. K. (1995). Quasi-linear geostatistical theory for inversing. *Water Resour. Res.*, 31(10):2411–2419.

Kitanidis, P. K. (1996). Analytical expressions of conditional mean, covariance, and sample functions in geostatistics. *Stoch. Hydrol. Hydraul.*, 12:279–294.

Kitanidis, P. K. (1997). *Introduction to Geostatistics*. Cambridge University Press, Cambridge.

Kozintsev, B. (1999). *Computations with Gaussian random fields*. PhD thesis, Institute for Systems Research, University of Maryland.

Liu, Q. H. and Ngyuen, N. (1998). An accurate algorithm for nonuniform fast Fourier transforms(NUFFT's). *IEEE Microwave and Guided Wave Letters*, 8(1):18–20.

Newsam, G. N. and Dietrich, C. R. (1994). Bounds on the size of nonnegative definite circulant embeddings of positive definite Toeplitz matrices. *IEEE Transactions on Information Theory*, 40(4):1218–1220.

Nowak, W. (2005). *Geostatistical Methods for the Identification of Flow and Transport Parameters in Subsurface Flow*. PhD thesis, Institut für Wasserbau, Universität Stuttgart, http://elib.uni-stuttgart.de/opus/frontdoor.php?source_opus=2275.

Nowak, W. and Cirpka, O. A. (2004). A modified Levenberg-Marquardt algorithm for quasi-linear geostatistical inversing. *Adv. Water Resour.*, 27(7):737–750.

Nowak, W. and Cirpka, O. A. (2006). Geostatistical inference of conductivity and dispersion coefficients from hydraulic heads and tracer data. *Water Resour. Res.*, 42(W08416):doi:10.1029/2005WR004832.

Nowak, W., Tenkleve, S., and Cirpka, O. A. (2003). Efficient computation of linearized cross-covariance and auto-covariance matrices of interdependent quantities. *Math. Geol.*, 35(1):53–66.

Pegram, G. G. S. (2004). Spatial interpolation and mapping of rainfall (SIMAR) Vol.3: Data merging for rainfall map production. *Water Research Commision Report*, (1153/1/04).

Press, W. H., Teukolsky, B. P. F. S. A., and Vetterling, W. T. (1992). *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, Cambridge, 2nd edition.

Rino, C. L. (1970). The inversion of covariance matrices by Finite Fourier Transforms. *IEEE Transactions on Information Theory*, 16:230–232.

Schweppe, F. C. (1973). *Uncertain Dynamic Systems*. Prentice-Hall, Englewood Cliffs, NJ.

Shewchuk, J. R. (1994). An introduction to the conjugate gradient method without the agonizing pain. http://www.cs.berkeley.edu/∼jrs/.

Strang, G. (1986). A proposal for Toeplitz matrix calculations. *Stud. Appl. Amth.*, 74:171–176.

Trapp, G. E. (1973). Inverses of circulant matrices and block circulant matrices. *Kyungpook Math. J.*, 13(1):11–20.

van Loan, C. F. (1992). *Computational Frameworks for the Fast Fourier Transform*. SIAM Publications, Philadelphia, PA.

Varga, R. S. (1954). Eigenvalues of circulant matrices. *Pacific J. Math.*, 4:151–160.

Wesson, S. M. and Pegram, G. G. S. (2004). Radar rainfall image repair techniques. *Hydrological and Earth Systems Sciences*, 8(2):8220–234.

Zimmerman, D. L. (1989). Computationally exploitable structure of covariance matrices and generalized covariance matrices in spatial models. *J. Stat. Comput. Sim.*, 32(1/2):1–15.

Figures

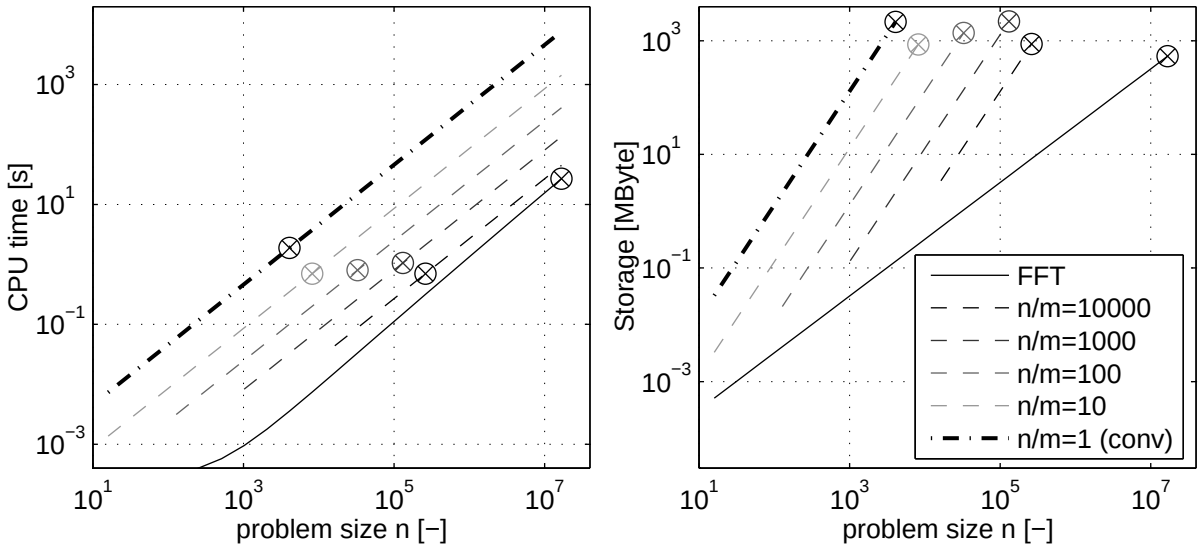
CPU time [s]
problem size n [−]
Storage [MByte]
problem size n [−]
FFT
n/m=10000
n/m=1000
n/m=100
n/m=10
n/m=1 (conv)

Figure 1:

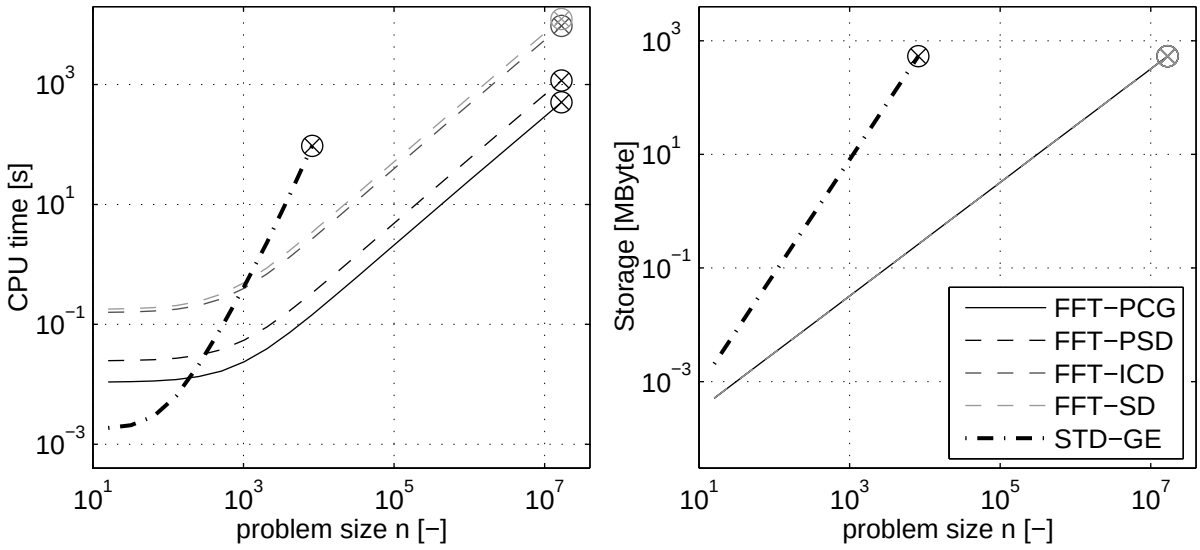
CPU time [s]
problem size n [−]
Storage [MByte]
problem size n [−]
FFT−PCG
FFT−PSD
FFT−ICD
FFT−SD
STD−GE

Figure 2:

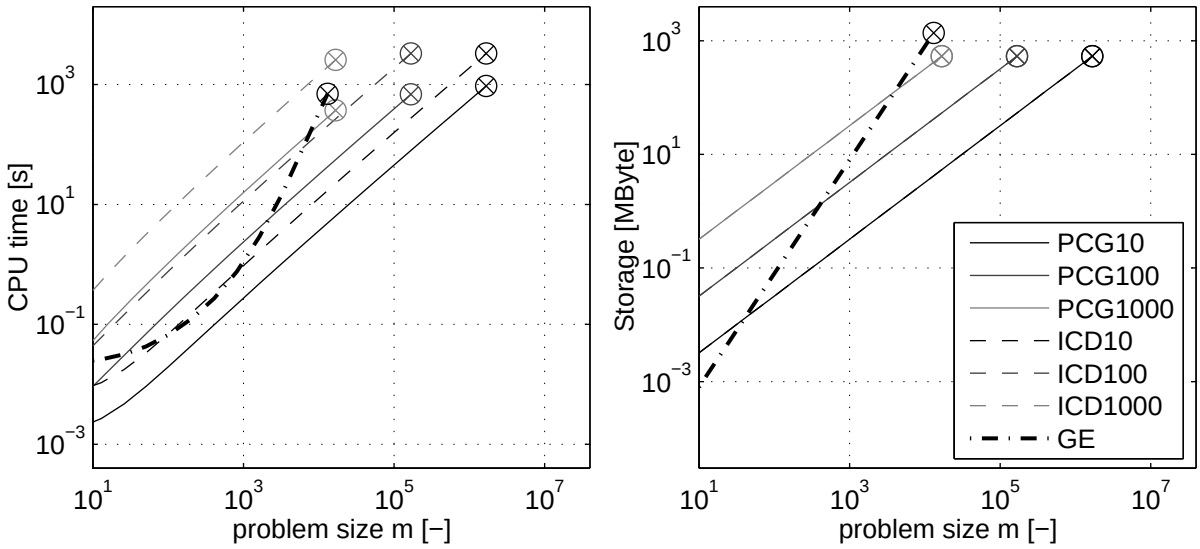
CPU time [s]
problem size m [−]
Storage [MByte]
problem size m [−]
PCG10
PCG100
PCG1000
ICD10
ICD100
ICD1000
GE

Figure 3:

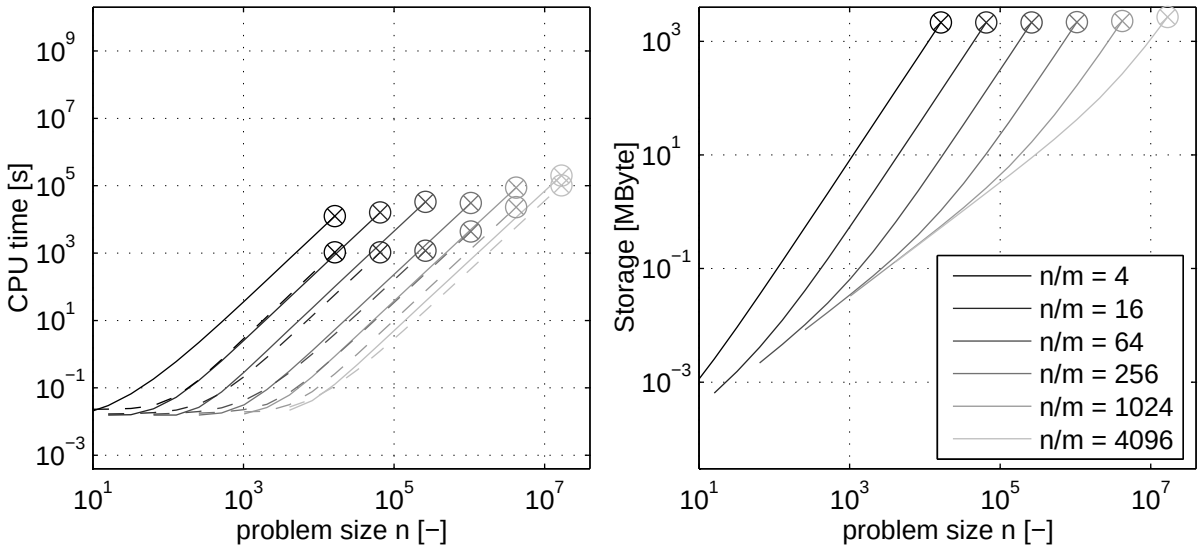
CPU time [s]
problem size n [−]
Storage [MByte]
problem size n [−]
n/m = 4
n/m = 16
n/m = 64
n/m = 256
n/m = 1024
n/m = 4096

Figure 4:

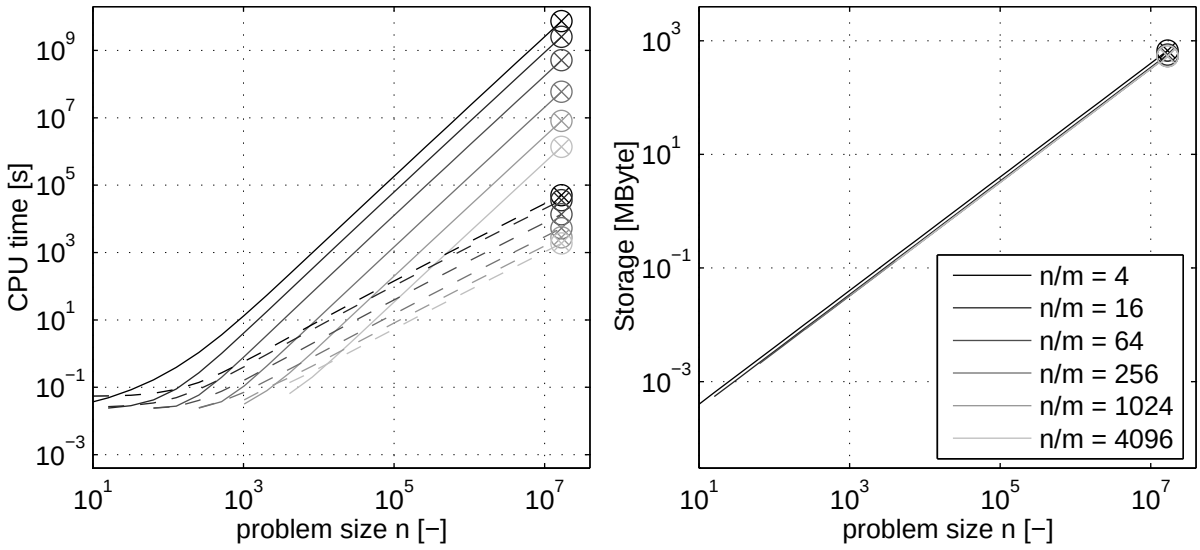
CPU time [s]
problem size n [−]
Storage [MByte]
problem size n [−]
n/m = 4
n/m = 16
n/m = 64
n/m = 256
n/m = 1024
n/m = 4096

Figure 5:

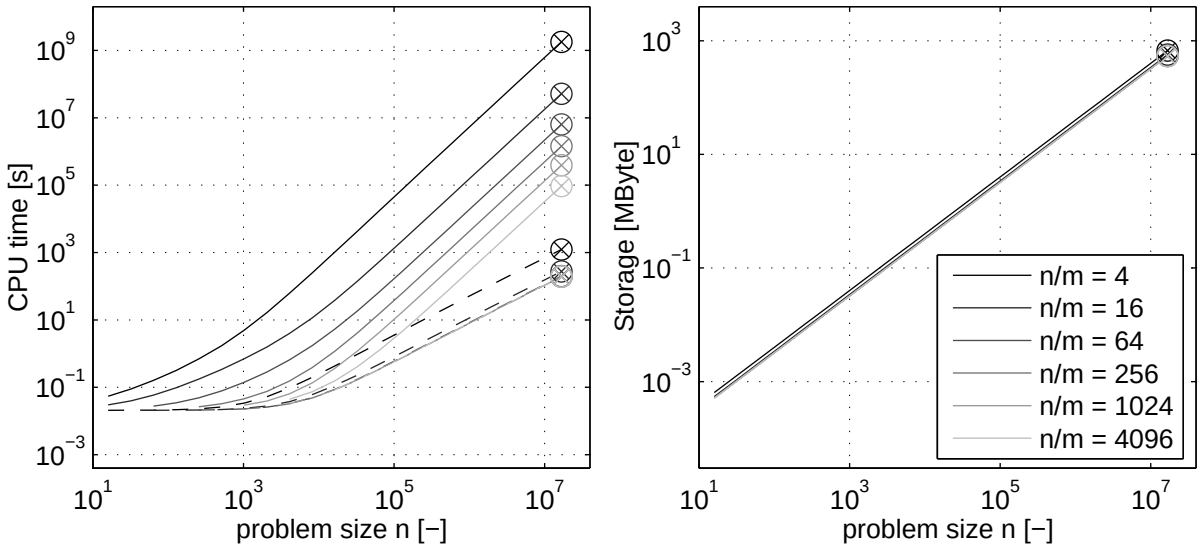
CPU time [s]
problem size n [−]
Storage [MByte]
problem size n [−]
n/m = 4
n/m = 16
n/m = 64
n/m = 256
n/m = 1024
n/m = 4096

Figure 6:

Captions:

Figure 1: Convolution/Superposition: CPU time (left) and storage requirements (right) for conventional discrete superposition versus superposition via FFT for different problem sizes n and numbers of superimposed terms m. Lines: fitted complexity models. X-Circles: memory overflow. Solid line: superposition/convolution via FFT (same for both tasks). Dashed lines: superposition via brute-force matrix product at different ratios n/m. Dash-dotted line: convolution via matrix product($n = m$). Lines exceeding point of memory overflow: superposition/convolution via successive addition of shifted kernel function. Lower line ends: minimum of $m = 1$ superimposed term.

Figure 2: Solvers for regular measurement grids: CPU time (left) and storage requirements (right) of different solution techniques for Toeplitz systems for different problem sizes m. Lines: fitted complexity models. X-Circles: memory overflow. FFT-PCG: FFT-based PCG solver. FFT-PSD: FFT-based Preconditioned Steepest Descent. FFT-ICD: Iterative Constrained Deconvolution. FFT-SD: FFT-based Steepest Descent. STD-GE: Standard Gaussian Elimination.

Figure 3: Extension to irregular measurement grids: CPU time (left) and storage requirements (right) of different solution techniques for nearly Toeplitz systems for different problem sizes m and different sizes n of the embedding Toeplitz matrix. Lines: fitted complexity models. X-Circles: memory overflow. PCG#: FFT-based PCG solver with # times larger regular grid. ICD#: FFT-based Iterative Constrained Deconvolution with # times larger regular grid. GE: Standard Gaussian Elimination.

Figure 4: Performance of Kriging with conventional solver: CPU time (left) and storage requirements (right) for different numbers n of estimation points and different numbers m of measurements. Lines: fitted complexity models. X-Circles: memory overflow. Solid lines: using conventional solver and conventional superposition. Dashed lines: using conventional solver and superposition via FFT.

Figure 5: Performance of Kriging with FFT-based PCG solver for irregularly scattered

measurements: CPU time (left) and storage requirements (right) for different numbers n of estimation points and different numbers m of measurements. Lines: fitted complexity models. X-Circles: memory overflow. Solid lines: exact estimation variance. Dashed lines: one-point approximation.

Figure 6: Performance of Kriging with FFT-based PCG-solver with measurements on regular grids: CPU time (left) and storage requirements (right) for different numbers n of estimation points and different numbers m of measurements. Lines: fitted complexity models. X-Circles: memory overflow. Solid lines: exact estimation variance. Dashed lines: large-grid approximation.