NAG Library Function Document

nag_dpttrf (f07jdc)

1 Purpose
nag_dpttrf (f07jdc) computes the modified Cholesky factorization of a real $n$ by $n$ symmetric positive definite tridiagonal matrix $A$.

2 Specification
#include <nag.h>
#include <nagf07.h>
void nag_dpttrf (Integer n, double d[], double e[], NagError *fail)

3 Description
nag_dpttrf (f07jdc) factorizes the matrix $A$ as

$$A = LDL^T,$$

where $L$ is a unit lower bidiagonal matrix and $D$ is a diagonal matrix with positive diagonal elements. The factorization may also be regarded as having the form $U^TDU$, where $U$ is a unit upper bidiagonal matrix.

4 References
None.

5 Arguments

1: $n$ – Integer $\hspace{1cm} \text{Input}$

   On entry: $n$, the order of the matrix $A$.

   Constraint: $n \geq 0$.

2: $d[\text{dim}]$ – double $\hspace{1cm} \text{Input/Output}$

   Note: the dimension, $\text{dim}$, of the array $d$ must be at least max$(1,n)$.

   On entry: must contain the $n$ diagonal elements of the matrix $A$.

   On exit: is overwritten by the $n$ diagonal elements of the diagonal matrix $D$ from the $LDL^T$ factorization of $A$.

3: $e[\text{dim}]$ – double $\hspace{1cm} \text{Input/Output}$

   Note: the dimension, $\text{dim}$, of the array $e$ must be at least max$(1,n-1)$.

   On entry: must contain the $(n-1)$ subdiagonal elements of the matrix $A$.

   On exit: is overwritten by the $(n-1)$ subdiagonal elements of the lower bidiagonal matrix $L$. ($e$ can also be regarded as containing the $(n-1)$ superdiagonal elements of the upper bidiagonal matrix $U$.)

4: $\text{fail}$ – NagError $\ast$ $\hspace{1cm} \text{Input/Output}$

   The NAG error argument (see Section 3.6 in the Essential Introduction).
6 Error Indicators and Warnings

**NE_ALLOC_FAIL**
Dynamic memory allocation failed.
See Section 3.2.1.2 in the Essential Introduction for further information.

**NE_BAD_PARAM**
On entry, argument `<value>` had an illegal value.

**NE_INT**
On entry, \(n = \langle \text{value} \rangle\).
Constraint: \(n \geq 0\).

**NE_INTERNAL_ERROR**
An internal error has occurred in this function. Check the function call and any array sizes. If the call is correct then please contact NAG for assistance.
An unexpected error has been triggered by this function. Please contact NAG.
See Section 3.6.6 in the Essential Introduction for further information.

**NE_MAT_NOT_POS_DEF**
The leading minor of order \(n\) is not positive definite, the factorization was completed, but \(d[n - 1] \leq 0\).
The leading minor of order \(<value>\) is not positive definite, the factorization could not be completed.

**NE_NO_LICENCE**
Your licence key may have expired or may not have been installed correctly.
See Section 3.6.5 in the Essential Introduction for further information.

7 Accuracy
The computed factorization satisfies an equation of the form
\[
A + E = LDL^T,
\]
where
\[
\|E\|_\infty = O(\epsilon)\|A\|_\infty
\]
and \(\epsilon\) is the *machine precision*.
Following the use of this function, nag_dpttrs (f07jec) can be used to solve systems of equations \(AX = B\), and nag_dptcon (f07jgc) can be used to estimate the condition number of \(A\).

8 Parallelism and Performance
Not applicable.

9 Further Comments
The total number of floating-point operations required to factorize the matrix \(A\) is proportional to \(n\).
The complex analogue of this function is nag_zpttrf (f07jrc).
10 Example

This example factorizes the symmetric positive definite tridiagonal matrix $A$ given by

$$
A = \begin{pmatrix}
4.0 & -2.0 & 0 & 0 & 0 \\
-2.0 & 10.0 & -6.0 & 0 & 0 \\
0 & -6.0 & 29.0 & 15.0 & 0 \\
0 & 0 & 15.0 & 25.0 & 8.0 \\
0 & 0 & 0 & 8.0 & 5.0 \\
\end{pmatrix}.
$$

10.1 Program Text

/* nag_dpttrf (f07jdc) Example Program. */
/* Copyright 2014 Numerical Algorithms Group. */
/* Mark 23, 2011. */

#include <stdio.h>
#include <nag.h>
#include <nag_stdlib.h>
#include <nagf07.h>

int main(void)
{
    /* Scalars */
    Integer exit_status = 0, i, n;
    /* Arrays */
    double *d = 0, *e = 0;
    /* Nag Types */
    NagError fail;
    INIT_FAIL(fail);
    printf("nag_dpttrf (f07jdc) Example Program Results\n\n");
    /* Skip heading in data file */
    #ifdef _WIN32
        scanf_s("%[\n]");
    #else
        scanf("%[\n]");
    #endif
    #ifdef _WIN32
        scanf_s("%"NAG_IFMT"%[\n]", &n);
    #else
        scanf("%"NAG_IFMT"%[\n]", &n);
    #endif
    if (n < 0)
    {
        printf("Invalid n\n");
        exit_status = 1;
        goto END;
    }
    /* Allocate memory */
    if (!(d = NAG_ALLOC(n, double)) || (!e = NAG_ALLOC(n-1, double)))
    {
        printf("Allocation failure\n");
        exit_status = -1;
        goto END;
    }
    /* Read the lower bidiagonal part of the tridiagonal matrix A from
     * data file

Mark 25

f07jdc.3
/*
 * Factorize the tridiagonal matrix A using nag_dpttrf (f07jdc).
 * nag_dpttrf(n, d, e, &fail);
 * if (fail.code != NE_NOERROR)
 *  { printf("Error from nag_dpttrf (f07jdc).\n%s\n", fail.message);
 *    exit_status = 1;
 *    goto END;
 * }
 *
 * Print details of the factorization */
 * printf("Details of factorization\n\n");
 * printf(" The diagonal elements of D\n");
 * for (i = 0; i < n; ++i) printf("%9.4f\n", d[i], i%8 == 7?"\n":" ");
 * printf("\n Sub-diagonal elements of the Cholesky factor L\n");
 * for (i = 0; i < n-1; ++i) printf("%9.4f\n", e[i], i%8 == 7?"\n":" ");
 *
 * END:
 * NAG_FREE(d);
 * NAG_FREE(e);
 * return exit_status;
 */

10.2 Program Data

nag_dpttrf (f07jdc) Example Program Data

5 : n
4.0 10.0 29.0 25.0 5.0 : diagonal D
-2.0 -6.0 15.0 8.0 : sub-diagonal E

10.3 Program Results

nag_dpttrf (f07jdc) Example Program Results

Details of factorization

The diagonal elements of D
4.0000 9.0000 25.0000 16.0000 1.0000

Sub-diagonal elements of the Cholesky factor L
-0.5000 -0.6667 0.6000 0.5000