

NAG Toolbox for MATLAB

d06cb

1 Purpose

d06cb generates the sparsity pattern of a finite element matrix associated with a given mesh.

2 Syntax

```
[nnz, irow, icol, ifail] = d06cb(nv, nnzmax, conn, 'nelt', nelt)
```

3 Description

d06cb generates the sparsity pattern of a finite element matrix associated with a given mesh. The sparsity pattern is returned in a co-ordinate storage format consistent with the sparse linear algebra functions in Chapter F11. More precisely d06cb returns the number of nonzero elements in the associated sparse matrix, and their row and column indices. This is designed to assist you in applying finite element discretization to meshes from the D06 Chapter Introduction and in solving the resulting sparse linear system using functions from Chapter F11.

The output sparsity pattern is based on the fact that finite element matrix A has elements a_{ij} satisfying:

$$a_{ij} \neq 0 \Rightarrow i \text{ and } j \text{ are vertices belonging to the same triangle.}$$

4 References

None.

5 Parameters

5.1 Compulsory Input Parameters

- 1: **nv – int32 scalar**
the total number of vertices in the input mesh.
Constraint: $nv \geq 3$.
- 2: **nnzmax – int32 scalar**
the maximum number of nonzero entries in the matrix based on the input mesh. It is the dimension of the arrays **irow** and **icol** as declared in the (sub)program from which d06cb is called.
Constraint: $4 \times \text{nelt} + nv \leq \text{nnzmax} \leq nv^2$.
- 3: **conn(3,nelt) – int32 array**
The connectivity of the mesh between triangles and vertices. For each triangle j , **conn**(i,j) gives the indices of its three vertices (in anticlockwise order), for $i = 1, 2, 3$ and $j = 1, \dots, \text{nelt}$.
Constraint: $1 \leq \text{conn}(i, j) \leq nv$ and $\text{conn}(1, j) \neq \text{conn}(2, j)$ and $\text{conn}(1, j) \neq \text{conn}(3, j)$ and $\text{conn}(2, j) \neq \text{conn}(3, j)$, for $i = 1, 2, 3$ and $j = 1, 2, \dots, \text{nelt}$.

5.2 Optional Input Parameters

- 1: **nelt – int32 scalar**
Default: The dimension of the array **conn**.

the number of triangles in the input mesh.

Constraint: $\text{nelt} \leq 2 \times \text{nv} - 1$.

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4 Output Parameters

1: **nnz** – int32 scalar

The number of nonzero entries in the matrix associated with the input mesh.

2: **irow(nnzmax)** – int32 array

3: **icol(nnzmax)** – int32 array

The first **nnz** elements contain the row and column indices of the nonzero elements supplied in the finite element matrix A .

4: **ifail** – int32 scalar

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail = 1

On entry, **nv** < 3,
 or **nelt** > $2 \times \text{nv} - 1$,
 or **nnzmax** < $4 \times \text{nelt} + \text{nv}$ or **nnzmax** > nv^2
 or **conn**(i,j) < 1 or **conn**(i,j) > **nv** for some $i = 1, 3$ and j , $1 \leq j \leq \text{nelt}$,
 or **conn**(1, j) = **conn**(2, j) or **conn**(1, j) = **conn**(3, j) or
conn(2, j) = **conn**(3, j) for some $j = 1, \dots, \text{nelt}$.

ifail = 2

A serious error has occurred in an internal call to an auxiliary function. Check the input mesh, especially the connectivity between triangles and vertices (the parameter **conn**). Array dimensions should be checked as well. If the problem persists, contact NAG.

7 Accuracy

Not applicable.

8 Further Comments

Not applicable.

9 Example

```
nv = int32(250);
nnzmax = int32(3000);
conn = zeros(3, 402, 'int32');
conn(1, 1:402) = [int32(21),
int32(17),
int32(54),
```

```
int32(148),
int32(57),
int32(58),
int32(22),
int32(205),
int32(20),
int32(147),
int32(70),
int32(23),
int32(146),
int32(59),
int32(245),
int32(52),
int32(145),
int32(180),
int32(101),
int32(127),
int32(19),
int32(144),
int32(129),
int32(51),
int32(143),
int32(61),
int32(24),
int32(137),
int32(18),
int32(142),
int32(72),
int32(182),
int32(207),
int32(62),
int32(211),
int32(210),
int32(140),
int32(63),
int32(114),
int32(145),
int32(111),
int32(139),
int32(90),
int32(122),
int32(138),
int32(118),
int32(209),
int32(214),
int32(113),
int32(132),
int32(83),
int32(136),
int32(80),
int32(68),
int32(117),
int32(175),
int32(79),
int32(200),
int32(117),
int32(134),
int32(78),
int32(64),
int32(127),
int32(27),
int32(64),
int32(133),
int32(86),
int32(48),
int32(134),
int32(129),
int32(116),
int32(131),
int32(121),
```

```
int32(66),  
int32(87),  
int32(130),  
int32(76),  
int32(125),  
int32(88),  
int32(89),  
int32(67),  
int32(219),  
int32(65),  
int32(130),  
int32(131),  
int32(126),  
int32(226),  
int32(90),  
int32(225),  
int32(126),  
int32(30),  
int32(222),  
int32(127),  
int32(230),  
int32(66),  
int32(91),  
int32(75),  
int32(129),  
int32(65),  
int32(125),  
int32(132),  
int32(131),  
int32(127),  
int32(144),  
int32(121),  
int32(68),  
int32(240),  
int32(123),  
int32(15),  
int32(49),  
int32(92),  
int32(93),  
int32(216),  
int32(164),  
int32(85),  
int32(94),  
int32(123),  
int32(129),  
int32(245),  
int32(166),  
int32(213),  
int32(70),  
int32(62),  
int32(160),  
int32(137),  
int32(207),  
int32(111),  
int32(160),  
int32(121),  
int32(13),  
int32(147),  
int32(33),  
int32(118),  
int32(188),  
int32(212),  
int32(149),  
int32(116),  
int32(149),  
int32(178),  
int32(34),  
int32(115),  
int32(183),  
int32(171),
```

```
int32(154),  
int32(228),  
int32(114),  
int32(14),  
int32(63),  
int32(32),  
int32(113),  
int32(149),  
int32(26),  
int32(150),  
int32(112),  
int32(141),  
int32(62),  
int32(44),  
int32(115),  
int32(149),  
int32(24),  
int32(166),  
int32(110),  
int32(246),  
int32(60),  
int32(154),  
int32(109),  
int32(135),  
int32(23),  
int32(37),  
int32(108),  
int32(181),  
int32(61),  
int32(155),  
int32(173),  
int32(177),  
int32(25),  
int32(185),  
int32(106),  
int32(181),  
int32(102),  
int32(38),  
int32(105),  
int32(151),  
int32(60),  
int32(1),  
int32(104),  
int32(155),  
int32(24),  
int32(179),  
int32(103),  
int32(188),  
int32(59),  
int32(155),  
int32(102),  
int32(191),  
int32(23),  
int32(193),  
int32(101),  
int32(149),  
int32(100),  
int32(162),  
int32(33),  
int32(151),  
int32(195),  
int32(152),  
int32(114),  
int32(153),  
int32(116),  
int32(176),  
int32(33),  
int32(250),  
int32(42),  
int32(176),
```

```
int32(44),  
int32(157),  
int32(242),  
int32(158),  
int32(94),  
int32(236),  
int32(32),  
int32(160),  
int32(97),  
int32(161),  
int32(26),  
int32(162),  
int32(157),  
int32(163),  
int32(114),  
int32(168),  
int32(99),  
int32(165),  
int32(62),  
int32(249),  
int32(222),  
int32(167),  
int32(115),  
int32(168),  
int32(160),  
int32(169),  
int32(108),  
int32(190),  
int32(166),  
int32(171),  
int32(169),  
int32(172),  
int32(13),  
int32(173),  
int32(60),  
int32(174),  
int32(33),  
int32(175),  
int32(172),  
int32(40),  
int32(35),  
int32(221),  
int32(204),  
int32(178),  
int32(61),  
int32(190),  
int32(176),  
int32(180),  
int32(173),  
int32(181),  
int32(10),  
int32(182),  
int32(24),  
int32(183),  
int32(156),  
int32(184),  
int32(181),  
int32(185),  
int32(38),  
int32(186),  
int32(184),  
int32(187),  
int32(1),  
int32(188),  
int32(151),  
int32(189),  
int32(24),  
int32(190),  
int32(156),  
int32(191),
```

```
int32(6),  
int32(192),  
int32(3),  
int32(193),  
int32(191),  
int32(194),  
int32(193),  
int32(195),  
int32(70),  
int32(196),  
int32(23),  
int32(238),  
int32(68),  
int32(198),  
int32(145),  
int32(199),  
int32(180),  
int32(236),  
int32(134),  
int32(201),  
int32(223),  
int32(202),  
int32(207),  
int32(203),  
int32(171),  
int32(204),  
int32(172),  
int32(205),  
int32(178),  
int32(206),  
int32(189),  
int32(207),  
int32(72),  
int32(212),  
int32(182),  
int32(209),  
int32(147),  
int32(210),  
int32(27),  
int32(234),  
int32(215),  
int32(212),  
int32(152),  
int32(213),  
int32(136),  
int32(214),  
int32(115),  
int32(215),  
int32(213),  
int32(216),  
int32(215),  
int32(217),  
int32(214),  
int32(218),  
int32(157),  
int32(231),  
int32(218),  
int32(220),  
int32(67),  
int32(221),  
int32(175),  
int32(222),  
int32(200),  
int32(223),  
int32(218),  
int32(224),  
int32(235),  
int32(225),  
int32(157),  
int32(226),
```

```

int32(133),
int32(227),
int32(48),
int32(228),
int32(134),
int32(229),
int32(123),
int32(230),
int32(67),
int32(234),
int32(219),
int32(232),
int32(72),
int32(233),
int32(226),
int32(234),
int32(211),
int32(235),
int32(30),
int32(236),
int32(150),
int32(237),
int32(128),
int32(238),
int32(230),
int32(239),
int32(131),
int32(240),
int32(218),
int32(241),
int32(74),
int32(242),
int32(133),
int32(243),
int32(162),
int32(244),
int32(220),
int32(245),
int32(213),
int32(246),
int32(164),
int32(247),
int32(94),
int32(248),
int32(129),
int32(249),
int32(162),
int32(250),
int32(151)];
conn(2, 1:402) = [ int32(55),
int32(217),
int32(55),
int32(119),
int32(21),
int32(20),
int32(55),
int32(165),
int32(21),
int32(119),
int32(250),
int32(53),
int32(151),
int32(19),
int32(213),
int32(53),
int32(240),
int32(60),
int32(196),
int32(224),
int32(20),

```

```
int32(46),
int32(230),
int32(52),
int32(135),
int32(165),
int32(169),
int32(142),
int32(19),
int32(16),
int32(73),
int32(25),
int32(100),
int32(113),
int32(215),
int32(26),
int32(112),
int32(81),
int32(110),
int32(49),
int32(148),
int32(122),
int32(138),
int32(163),
int32(16),
int32(137),
int32(83),
int32(137),
int32(82),
int32(240),
int32(84),
int32(120),
int32(81),
int32(244),
int32(84),
int32(177),
int32(80),
int32(30),
int32(118),
int32(243),
int32(79),
int32(77),
int32(133),
int32(28),
int32(78),
int32(127),
int32(87),
int32(132),
int32(223),
int32(131),
int32(163),
int32(241),
int32(16),
int32(130),
int32(16),
int32(126),
int32(77),
int32(75),
int32(16),
int32(138),
int32(197),
int32(120),
int32(75),
int32(129),
int32(129),
int32(75),
int32(133),
int32(15),
int32(128),
int32(131),
int32(200),
```

```
int32(200),  
int32(200),  
int32(67),  
int32(129),  
int32(15),  
int32(131),  
int32(202),  
int32(125),  
int32(76),  
int32(48),  
int32(232),  
int32(237),  
int32(201),  
int32(123),  
int32(69),  
int32(48),  
int32(142),  
int32(16),  
int32(153),  
int32(15),  
int32(158),  
int32(112),  
int32(141),  
int32(121),  
int32(95),  
int32(121),  
int32(66),  
int32(122),  
int32(45),  
int32(120),  
int32(41),  
int32(209),  
int32(247),  
int32(118),  
int32(164),  
int32(147),  
int32(95),  
int32(85),  
int32(14),  
int32(117),  
int32(34),  
int32(84),  
int32(6),  
int32(110),  
int32(160),  
int32(122),  
int32(97),  
int32(115),  
int32(35),  
int32(18),  
int32(187),  
int32(114),  
int32(35),  
int32(223),  
int32(50),  
int32(15),  
int32(113),  
int32(33),  
int32(81),  
int32(168),  
int32(140),  
int32(174),  
int32(216),  
int32(143),  
int32(147),  
int32(45),  
int32(205),  
int32(164),  
int32(25),  
int32(162),
```

```
int32(114),
int32(141),
int32(107),
int32(156),
int32(19),
int32(151),
int32(189),
int32(38),
int32(52),
int32(186),
int32(205),
int32(250),
int32(60),
int32(151),
int32(112),
int32(39),
int32(52),
int32(177),
int32(105),
int32(39),
int32(20),
int32(155),
int32(104),
int32(2),
int32(19),
int32(192),
int32(206),
int32(42),
int32(53),
int32(192),
int32(102),
int32(183),
int32(19),
int32(188),
int32(101),
int32(191),
int32(54),
int32(98),
int32(164),
int32(166),
int32(150),
int32(135),
int32(250),
int32(116),
int32(152),
int32(49),
int32(153),
int32(35),
int32(154),
int32(151),
int32(179),
int32(36),
int32(170),
int32(124),
int32(237),
int32(15),
int32(158),
int32(150),
int32(159),
int32(14),
int32(160),
int32(112),
int32(161),
int32(144),
int32(237),
int32(116),
int32(212),
int32(172),
int32(164),
int32(147),
```

```
int32(165),  
int32(45),  
int32(170),  
int32(17),  
int32(167),  
int32(164),  
int32(168),  
int32(110),  
int32(206),  
int32(156),  
int32(170),  
int32(51),  
int32(171),  
int32(11),  
int32(172),  
int32(109),  
int32(173),  
int32(156),  
int32(174),  
int32(10),  
int32(175),  
int32(187),  
int32(176),  
int32(181),  
int32(177),  
int32(107),  
int32(178),  
int32(42),  
int32(185),  
int32(18),  
int32(180),  
int32(8),  
int32(221),  
int32(110),  
int32(182),  
int32(155),  
int32(183),  
int32(7),  
int32(184),  
int32(183),  
int32(185),  
int32(6),  
int32(186),  
int32(183),  
int32(187),  
int32(191),  
int32(188),  
int32(108),  
int32(189),  
int32(43),  
int32(190),  
int32(4),  
int32(191),  
int32(187),  
int32(194),  
int32(3),  
int32(193),  
int32(192),  
int32(194),  
int32(146),  
int32(195),  
int32(53),  
int32(196),  
int32(135),  
int32(220),  
int32(48),  
int32(198),  
int32(107),  
int32(199),  
int32(30),
```

```
int32(237),  
int32(46),  
int32(243),  
int32(143),  
int32(232),  
int32(52),  
int32(203),  
int32(135),  
int32(246),  
int32(111),  
int32(205),  
int32(169),  
int32(206),  
int32(141),  
int32(207),  
int32(112),  
int32(208),  
int32(82),  
int32(209),  
int32(140),  
int32(211),  
int32(27),  
int32(211),  
int32(208),  
int32(212),  
int32(139),  
int32(240),  
int32(148),  
int32(214),  
int32(122),  
int32(215),  
int32(140),  
int32(216),  
int32(17),  
int32(217),  
int32(136),  
int32(228),  
int32(120),  
int32(219),  
int32(146),  
int32(220),  
int32(177),  
int32(221),  
int32(150),  
int32(222),  
int32(47),  
int32(228),  
int32(133),  
int32(224),  
int32(28),  
int32(234),  
int32(28),  
int32(226),  
int32(132),  
int32(227),  
int32(218),  
int32(228),  
int32(16),  
int32(229),  
int32(143),  
int32(230),  
int32(211),  
int32(231),  
int32(202),  
int32(241),  
int32(128),  
int32(233),  
int32(124),  
int32(234),  
int32(200),
```

```
int32(235),  
int32(31),  
int32(236),  
int32(200),  
int32(237),  
int32(197),  
int32(238),  
int32(202),  
int32(239),  
int32(145),  
int32(240),  
int32(232),  
int32(241),  
int32(128),  
int32(242),  
int32(201),  
int32(243),  
int32(69),  
int32(244),  
int32(153),  
int32(245),  
int32(204),  
int32(246),  
int32(14),  
int32(247),  
int32(66),  
int32(248),  
int32(46),  
int32(249),  
int32(41),  
int32(250)];  
conn(3, 1:402) = [ int32(56),  
int32(137),  
int32(22),  
int32(118),  
int32(56),  
int32(57),  
int32(21),  
int32(111),  
int32(57),  
int32(148),  
int32(195),  
int32(196),  
int32(197),  
int32(102),  
int32(49),  
int32(103),  
int32(198),  
int32(199),  
int32(54),  
int32(235),  
int32(105),  
int32(201),  
int32(202),  
int32(203),  
int32(204),  
int32(205),  
int32(206),  
int32(17),  
int32(109),  
int32(17),  
int32(241),  
int32(208),  
int32(71),  
int32(209),  
int32(210),  
int32(211),  
int32(161),  
int32(113),  
int32(212),
```

```
int32(213),  
int32(214),  
int32(215),  
int32(89),  
int32(216),  
int32(88),  
int32(148),  
int32(117),  
int32(217),  
int32(209),  
int32(218),  
int32(117),  
int32(219),  
int32(63),  
int32(220),  
int32(118),  
int32(221),  
int32(63),  
int32(236),  
int32(119),  
int32(223),  
int32(63),  
int32(78),  
int32(224),  
int32(225),  
int32(63),  
int32(242),  
int32(121),  
int32(227),  
int32(228),  
int32(126),  
int32(122),  
int32(74),  
int32(229),  
int32(65),  
int32(121),  
int32(65),  
int32(64),  
int32(76),  
int32(87),  
int32(88),  
int32(238),  
int32(231),  
int32(125),  
int32(126),  
int32(239),  
int32(65),  
int32(233),  
int32(138),  
int32(234),  
int32(75),  
int32(235),  
int32(236),  
int32(237),  
int32(238),  
int32(130),  
int32(90),  
int32(74),  
int32(239),  
int32(64),  
int32(64),  
int32(240),  
int32(241),  
int32(242),  
int32(243),  
int32(118),  
int32(244),  
int32(198),  
int32(137),  
int32(138),
```

```
int32(245),  
int32(91),  
int32(92),  
int32(140),  
int32(246),  
int32(118),  
int32(247),  
int32(229),  
int32(248),  
int32(139),  
int32(249),  
int32(240),  
int32(250),  
int32(147),  
int32(95),  
int32(123),  
int32(100),  
int32(148),  
int32(96),  
int32(86),  
int32(160),  
int32(119),  
int32(154),  
int32(85),  
int32(186),  
int32(208),  
int32(97),  
int32(245),  
int32(98),  
int32(107),  
int32(154),  
int32(199),  
int32(192),  
int32(51),  
int32(156),  
int32(132),  
int32(51),  
int32(158),  
int32(62),  
int32(159),  
int32(82),  
int32(160),  
int32(161),  
int32(170),  
int32(163),  
int32(204),  
int32(165),  
int32(166),  
int32(214),  
int32(168),  
int32(182),  
int32(170),  
int32(171),  
int32(204),  
int32(199),  
int32(174),  
int32(104),  
int32(177),  
int32(103),  
int32(176),  
int32(103),  
int32(184),  
int32(178),  
int32(179),  
int32(180),  
int32(186),  
int32(208),  
int32(187),  
int32(108),  
int32(186),
```

```
int32(58),
int32(185),
int32(58),
int32(188),
int32(59),
int32(192),
int32(59),
int32(188),
int32(189),
int32(190),
int32(23),
int32(194),
int32(58),
int32(192),
int32(105),
int32(194),
int32(22),
int32(194),
int32(22),
int32(99),
int32(99),
int32(249),
int32(159),
int32(197),
int32(146),
int32(50),
int32(50),
int32(50),
int32(50),
int32(36),
int32(174),
int32(146),
int32(250),
int32(37),
int32(190),
int32(234),
int32(128),
int32(92),
int32(93),
int32(222),
int32(31),
int32(247),
int32(96),
int32(25),
int32(25),
int32(243),
int32(134),
int32(152),
int32(152),
int32(13),
int32(149),
int32(111),
int32(61),
int32(46),
int32(162),
int32(18),
int32(18),
int32(172),
int32(13),
int32(171),
int32(106),
int32(179),
int32(44),
int32(203),
int32(106),
int32(12),
int32(12),
int32(104),
int32(104),
int32(170),
```

```
int32(150),  
int32(11),  
int32(11),  
int32(39),  
int32(156),  
int32(9),  
int32(175),  
int32(60),  
int32(60),  
int32(43),  
int32(156),  
int32(109),  
int32(109),  
int32(9),  
int32(9),  
int32(169),  
int32(169),  
int32(179),  
int32(179),  
int32(8),  
int32(8),  
int32(156),  
int32(176),  
int32(7),  
int32(7),  
int32(185),  
int32(40),  
int32(6),  
int32(186),  
int32(103),  
int32(23),  
int32(44),  
int32(170),  
int32(5),  
int32(5),  
int32(1),  
int32(2),  
int32(4),  
int32(4),  
int32(2),  
int32(3),  
int32(244),  
int32(69),  
int32(54),  
int32(101),  
int32(143),  
int32(67),  
int32(49),  
int32(49),  
int32(115),  
int32(18),  
int32(31),  
int32(162),  
int32(47),  
int32(47),  
int32(141),  
int32(141),  
int32(106),  
int32(106),  
int32(177),  
int32(175),  
int32(214),  
int32(115),  
int32(106),  
int32(108),  
int32(164),  
int32(71),  
int32(163),  
int32(110),  
int32(83),
```

```
int32(117),
int32(26),
int32(26),
int32(225),
int32(120),
int32(112),
int32(163),
int32(215),
int32(120),
int32(137),
int32(167),
int32(216),
int32(120),
int32(210),
int32(210),
int32(167),
int32(167),
int32(219),
int32(124),
int32(211),
int32(124),
int32(197),
int32(197),
int32(181),
int32(10),
int32(170),
int32(162),
int32(227),
int32(132),
int32(29),
int32(29),
int32(233),
int32(128),
int32(29),
int32(29),
int32(223),
int32(47),
int32(124),
int32(157),
int32(142),
int32(142),
int32(202),
int32(66),
int32(27),
int32(124),
int32(141),
int32(207),
int32(225),
int32(28),
int32(231),
int32(231),
int32(127),
int32(29),
int32(159),
int32(159),
int32(162),
int32(157),
int32(135),
int32(143),
int32(232),
int32(232),
int32(213),
int32(136),
int32(207),
int32(73),
int32(233),
int32(233),
int32(47),
int32(134),
int32(195),
```

```
int32(146),  
int32(116),  
int32(139),  
int32(175),  
int32(172),  
int32(158),  
int32(158),  
int32(230),  
int32(230),  
int32(144),  
int32(144),  
int32(42),  
int32(155)];  
[nnz, irow, icol, ifail] = d06cb(nv, nnzmax, conn, 'nelt', int32(402));
```
