

Fill-in di una matrice

Costruire la matrice A così definita ($n = 10$):

$$a_{ii} = 3 \quad \text{per } i = 2, \dots, n$$

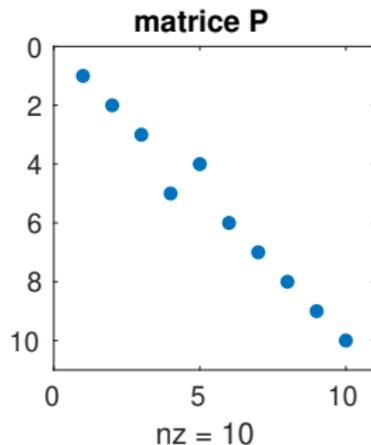
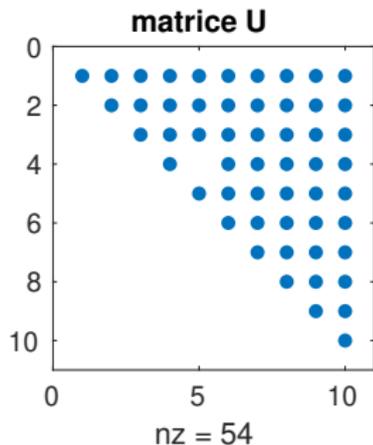
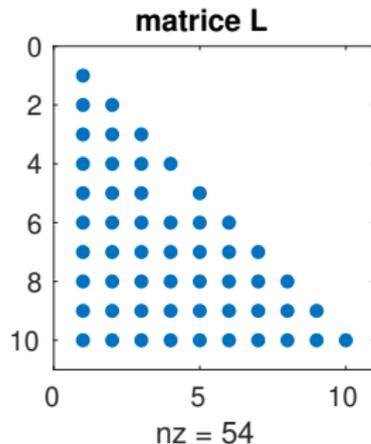
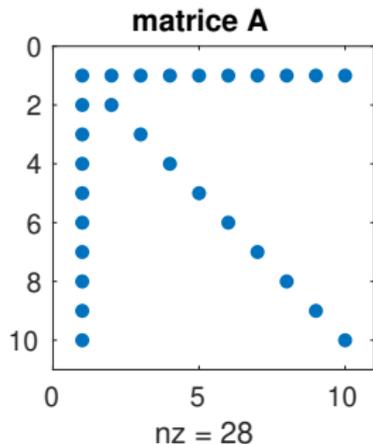
$$a_{1j} = 1 \quad \text{per } j = 1, \dots, n$$

$$a_{i1} = 1 \quad \text{per } i = 1, \dots, n$$

Richiamare la fattorizzazione LU con pivotazione e visualizzare il pattern di A , di L , di U e di P (matrice di pivotazione) con il comando `spy`:

```
A=[ . . .];  
figure(1); clf  
spy(A)  
[L,U,P]=lufact(A,1);
```

I pattern delle matrici



Fill-in di matrice

La fattorizzazione LU (ma anche il MEG) genera elementi non nulli in L e U anche dove il corrispondente elemento di A era nullo.

Questo fenomeno si chiama **fill-in**.

Anche se la matrice A ha pochi elementi non nulli, le matrici L e U possono avere una densità molto elevata.

Tutti i metodi diretti agiscono sulla matrice, trasformandola in una matrice più densa.

Al contrario i metodi iterativi non modificano la matrice del sistema, quindi quando si utilizzano i metodi iterativi non si verifica il fill-in.

Come evitare il fill-in

Esistono metodi di riordinamento di righe e colonne di una matrice da applicare prima di svolgere la fattorizzazione LU o il MEG con lo scopo di minimizzare il fill-in.

Esempio: `colperm` `colperm` costruisce una permutazione delle colonne in modo da spostare le righe e colonne di A con il maggior numero di elementi verso il fondo della matrice.

```
A = . . . % stessa matrice di prima
q = colperm(A); % q e' un vettore che contiene
% una permutazione delle colonne
Ap = A(q, q); % matrice permutata
spy(Ap)
[L, U, P] = lufact(Ap, 1);
```

I pattern delle matrici dopo la permutazione

