Lezioni di Calcolo Numerico
Lezione 07: Sistemi lineari

Alberto Tibaldi

5 maggio 2018

Indice

3 Metodo delle eliminazioni di Gauss

Se dovessimo rispondere alla domanda

> «Dovendo risolvere un generico sistema lineare, quali algoritmi conosciamo?»

La risposta, probabilmente, sarebbe **la regola di Cramer**[1]. Tuttavia[2] il calcolo del determinante, necessario per effettuare questa procedura, è estremamente pesante dal punto di vista computazionale. L'applicazione della regola di Laplace, infatti, ha un costo fattoriale, ossia circa $n!$ operazioni richieste: è totalmente inaccettabile per un computer! Tanto per capirci, immaginando che il nostro calcolatore sia in grado di lavorare ai PHz[3], la soluzione di un sistema con 24 equazioni in 24 incognite mediante regola di Cramer richiederebbe 50 anni. Questo, ci insegni la seguente regola del pollice: **gli algoritmi a costo esponenziale o peggio ancora fattoriale sono terribili e vanno evitati, sempre.**

Facciamo ora un passo indietro: nella precedente lezione abbiamo dedicato la nostra attenzione alle tecniche di soluzione di sistemi lineari triangolari, arrivando a trovare un costo computazionale circa pari a $n^2/2$.

> «Sì va beh ma cosa c'entra? Cioè, vogliamo dire che ora tutti i sistemi lineari sono triangolari?»

Beh, no: non è che capiti così spesso di aver a che fare con un sistema lineare *nativamente* triangolare. Tuttavia, la tecnica più diffusa per la soluzione di un generico sistema lineare si basa sul **trasformare il sistema iniziale in un sistema triangolare equivalente** per poi applicare le tecniche studiate nella precedente sezione. Se avessimo il nostro fantacalcolatore da 1 PHz, invece di 50 anni avremmo bisogno di 10^{-12} secondi. Un po' meglio, no? ;-)

Abbiamo appena nominato il concetto di **equivalenza** tra due sistemi lineari. In particolare, un sistema lineare è equivalente a un altro se, pur essendo questi diversi tra loro, presentano la stessa soluzione. Dall'Algebra Lineare, in effetti, dovrebbe essere noto che scambiando l'ordine di due righe della matrice e del termine noto, o riscrivendo un'equazione come una combinazione lineare di sé stessa e di un'altra equazione, il sistema lineare di partenza e quello così ottenuto sono **equivalenti**, ovvero, hanno la stessa soluzione. Basandoci su questo principio, proporremo ora un **algoritmo** atto a trasformare, mediante un approccio **sistematico**, un sistema lineare da generico a triangolare.

[1] quella basata sul calcolo dei determinanti, si veda per esempio https://it.wikipedia.org/wiki/Regola_di_Cramer

[2] a meno di furberie che introdurremo tra qualche sezione

[3] PHz sta per petahertz, ossia un milione di GHz, che significherebbe eseguire 10^{15} operazioni al secondo; per dare un'idea, i computer attuali lavorano a 2-3 GHz :-/

Poniamoci in particolare l'obiettivo di ottenere un sistema triangolare superiore a partire da un sistema generico. In questo senso il nostro algoritmo dovrà, per ogni colonna della matrice associata al sistema, effettuare un certo numero di operazioni atto a introdurre degli zeri al posto dei suoi elementi. Il metodo richiede un certo numero di passi, che dipende sostanzialmente dalla dimensione della matrice. Verrebbe dunque da pensare che il metodo che stiamo per studiare sia *iterativo*, poiché non si risolve *in un colpo solo*, ma **no**: si sa a priori quanti passi servono, quindi è un metodo diretto o, anzi, il **principe** dei metodi diretti!

Probabilmente, al fine di fissare le idee, la cosa migliore è investire una certa quantità di tempo su un esercizio per poi cercare di riassumere l'algoritmo in maniera più schematica. Il nostro obiettivo sarà dunque risolvere il sistema lineare

$$\underbrace{\begin{bmatrix} 2 & -1 & 1 & -2 \\ 0 & 2 & 0 & -1 \\ 1 & 0 & -2 & 1 \\ 0 & 2 & 1 & 1 \end{bmatrix}}_{\underline{\underline{A}}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}}_{\underline{x}} = \underbrace{\begin{bmatrix} 0 \\ 1 \\ 0 \\ 4 \end{bmatrix}}_{\underline{b}}.$$

Per questioni di notazione, diciamo che la prima operazione consiste semplicemente nel definire la matrice di sistema **all'inizio del primo passo** $\underline{\underline{A}}^{(1)}$, e il corrispondente termine noto $\underline{b}^{(1)}$, come quelli del sistema originale:

$$\underbrace{\begin{bmatrix} 2 & -1 & 1 & -2 \\ 0 & 2 & 0 & -1 \\ 1 & 0 & -2 & 1 \\ 0 & 2 & 1 & 1 \end{bmatrix}}_{\underline{\underline{A}}^{(1)}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}}_{\underline{x}} = \underbrace{\begin{bmatrix} 0 \\ 1 \\ 0 \\ 4 \end{bmatrix}}_{\underline{b}^{(1)}}.$$

Dal momento che la matrice è 4×4, avremo bisogno di 3 (ovvero $n-1$) passi per portare il sistema a triangolare. L'obiettivo del k-esimo passo è prendere la k-esima colonna e modificarla in modo tale che le sue componenti, dalla $(k+1)$-esima all'ultima, diventino 0. Per ogni k-esimo passo, toccheremo dunque tutte le righe a partire dalla $(k+1)$-esima fino all'ultima. Prima di tutto, guardiamo la matrice, e notiamo che $a_{kk}^{(k)}$, ossia, l'elemento alla posizione (k,k) della matrice di partenza al k-esimo passo, è diverso da 0; questo ci serve, perché tra poco dovremo calcolare dei coefficienti che contengono $a_{kk}^{(k)}$ al denominatore e quindi questo non può essere nullo. Nel caso questo fosse stato nullo, avremmo dovuto effettuare uno scambio di righe della matrice (e del termine noto) con la prima riga successiva avente $a_{ik}^{(k)}$ diverso da 0. Questa **dovrà** esistere, altrimenti la matrice avrebbe un'intera colonna piena di zeri, quindi avrebbe determinante uguale a 0 e la soluzione del sistema non sarebbe unica.

Detto questo, per ogni i-esima riga della k-esima colonna, calcoliamo un coefficiente m_{ik} definito come

$$m_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}.$$

In particolare, per $k = 1$, abbiamo:

$$m_{21} = 0; \qquad m_{31} = \frac{1}{2}; \qquad m_{41} = 0.$$

Questi m_{ik} sono i coefficienti della combinazione lineare che useremo per modificare ciascuna delle i-esime righe (sotto la k-esima). Il caso in cui m_{ik} è uguale a 0 è molto gradito, dal momento che significa *non toccare la i-esima riga* (è già a 0, quindi è già a posto). Per $k = 1$ quindi, in questo esempio, solo la terza riga va modificata.

La modifica alla i-esima riga consiste nel prendere ciascuna delle sue componenti e sottrarre la componente della k-esima riga sulla medesima colonna moltiplicata per m_{ik}. Considerando quindi la terza riga (l'unica legata a un coefficiente non nullo), l'operazione da fare sarà la seguente sostituzione:

$$\begin{bmatrix}1 & 0 & -2 & 1\end{bmatrix} \Longrightarrow \begin{bmatrix}1 & 0 & -2 & 1\end{bmatrix} - \frac{1}{2}\begin{bmatrix}2 & -1 & 1 & -2\end{bmatrix} = \begin{bmatrix}0 & -\left(\frac{1}{2}\right)(-1) & -2-\frac{1}{2} & 1+1\end{bmatrix},$$

dove abbiamo ottenuto il nostro obiettivo: abbiamo il primo elemento uguale a 0! Ovviamente, come già detto, anche il termine noto va modificato; in particolare,

$$b_3^{(2)} \Longrightarrow b_3^{(1)} - m_{31} b_1^{(1)},$$

che porta a ottenere

$$b_3^{(2)} = 0,$$

essendo che sia $b_1^{(1)}$ sia $b_3^{(1)}$ sono a 0. La **fortunata** (ma anche **fortuita**) situazione $b_k^{(k)} = 0$ è interessante perché permette di non sostituire **alcuna** componente del vettore, anche per le componenti aventi $m_{ik} \neq 0$. Per riassumere, dopo il passo $k = 1$, il sistema lineare è diventato

$$\underbrace{\begin{bmatrix}2 & -1 & 1 & -2\\ 0 & 2 & 0 & -1\\ 0 & \frac{1}{2} & -\frac{5}{2} & 2\\ 0 & 2 & 1 & 1\end{bmatrix}}_{\underline{\underline{A}}^{(2)}} \underbrace{\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix}}_{\underline{x}} = \underbrace{\begin{bmatrix}0\\ 1\\ 0\\ 4\end{bmatrix}}_{\underline{b}^{(2)}},$$

dove gli apici (2) indicano che queste matrici, ottenute dopo aver effettuato il primo passo, **sono il punto di partenza del passo 2**. Come scopriremo tra non molto, è utile tenere traccia dei coefficienti che abbiamo utilizzato; la notazione m_{ik} si presta infatti a inserirli in una matrice; iniziamo quindi a riempire una certa matrice $\underline{\underline{L}}$ contenente ciascuno di questi m_{ik}:

$$\underline{\underline{L}}^{(2)} = \begin{bmatrix}* & * & * & *\\ 0 & * & * & *\\ \frac{1}{2} & * & * & *\\ 0 & * & * & *\end{bmatrix}.$$

Il passo $k = 1$ è ora davvero completato. Possiamo procedere con il passo $k = 2$. A questo fine, scriviamo la matrice $\underline{\underline{A}}^{(1)}$ evidenziando alcuni elementi con diversi colori:

$$\begin{bmatrix}2 & -1 & 1 & -2\\ 0 & 2 & 0 & -1\\ 0 & \frac{1}{2} & -\frac{5}{2} & 2\\ 0 & 2 & 1 & 1\end{bmatrix}.$$

La prima colonna è già a posto: contiene tutti zeri! Dunque, dovremo preoccuparci di lavorare solo sulla sottomatrice 3×3 degli elementi evidenziati in blu. In dettaglio, il nostro obiettivo sarà far sì che la seconda colonna, dalla terza componente (inclusa) in poi, contenga solo zeri. Per far questo, procediamo analogamente a prima; prima di tutto, calcoliamo gli m_{ik}:

$$m_{32} = \frac{\frac{1}{2}}{2} = \frac{1}{4}; \qquad m_{42} = \frac{2}{2} = 1.$$

Dovremo sostituire, proprio come prima,

$$\begin{bmatrix}\frac{1}{2} & -\frac{5}{2} & -1\end{bmatrix} \Longrightarrow \begin{bmatrix}\frac{1}{2} & -\frac{5}{2} & -1\end{bmatrix} - \frac{1}{4}\begin{bmatrix}2 & 0 & -1\end{bmatrix} = \begin{bmatrix}0 & -\frac{5}{2}-0 & 2+\frac{1}{4}\end{bmatrix}$$

e

$$\begin{bmatrix}2 & 1 & 1\end{bmatrix} \Longrightarrow \begin{bmatrix}2 & 1 & 1\end{bmatrix} - 1\begin{bmatrix}2 & 0 & -1\end{bmatrix} = \begin{bmatrix}0 & 1 & 2\end{bmatrix},$$

che si possono includere nella sottomatrice 3×3 trasformando quella prima evidenziata in blu nella seguente:

$$\begin{bmatrix} 2 & 0 & -1 \\ 1 & -\frac{5}{2} & 2 \\ 2 & 1 & 1 \end{bmatrix} \Longrightarrow \begin{bmatrix} 2 & 0 & -1 \\ 0 & -\frac{5}{2} & \frac{9}{4} \\ 0 & 1 & 2 \end{bmatrix}.$$

Per quanto riguarda il termine noto, dovremo effettuare un'operazione simile; non avendo questa volta il fortuito caso $b_k^{(k)} = 0$, dovremo fare un conto analogo a quello appena svolto:

$$\underline{b}^{(3)} = \begin{bmatrix} 0 \\ 1 \\ 0 - \frac{1}{4} \\ 4 - 1 \end{bmatrix}.$$

Sostituendo la sottomatrice 3×3 nella parte di $\underline{\underline{A}}^{(1)}$ evidenziata in blu e usando questo $\underline{b}^{(2)}$, è possibile ottenere il risultato del secondo passo

$$\underbrace{\begin{bmatrix} 2 & -1 & 1 & -2 \\ 0 & 2 & 0 & -1 \\ 0 & 0 & -\frac{5}{2} & \frac{9}{4} \\ 0 & 0 & 1 & 2 \end{bmatrix}}_{\underline{\underline{A}}^{(3)}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}}_{\underline{x}} = \underbrace{\begin{bmatrix} 0 \\ 1 \\ -\frac{1}{4} \\ 3 \end{bmatrix}}_{\underline{b}^{(3)}},$$

e possiamo anche *mettere da parte* i moltiplicatori di questo secondo passo, aggiornando la matrice $\underline{\underline{L}}^{(2)}$ a $\underline{\underline{L}}^{(3)}$:

$$\underline{\underline{L}}^{(3)} = \begin{bmatrix} * & * & * & * \\ 0 & * & * & * \\ \frac{1}{2} & \frac{1}{4} & * & * \\ 0 & 1 & * & * \end{bmatrix}.$$

Questo decreta la fine del secondo passo. Possiamo ora quindi procedere con l'ultimo passo, $k = 3$. Per questo, rivediamo la matrice $\underline{\underline{A}}^{(3)}$, per capire cosa dobbiamo andare a toccare.

$$\begin{bmatrix} 2 & -1 & 1 & -2 \\ 0 & 2 & 0 & -1 \\ 0 & 0 & -\frac{5}{2} & \frac{9}{4} \\ 0 & 0 & 1 & 2 \end{bmatrix}.$$

La prima e la seconda colonna sono a posto. Rimane quindi da lavorare solamente sulla sottomatrice 2×2 evidenziata in blu. Per questo motivo, l'unico coefficiente da calcolare sarà

$$m_{43} = \frac{1}{-\frac{5}{2}} = -\frac{2}{5}.$$

Procedendo come prima, quindi, sostituiamo all'ultima riga sé stessa, meno la penultima moltiplicata per il coefficiente:

$$\begin{bmatrix} 1 & 2 \end{bmatrix} \Longrightarrow \begin{bmatrix} 1 & 2 \end{bmatrix} - \left(-\frac{2}{5}\right)\begin{bmatrix} -\frac{5}{2} & \frac{9}{4} \end{bmatrix} = \begin{bmatrix} 0 & \frac{29}{10} \end{bmatrix}.$$

Questa si può includere nella sottomatrice 2×2 trasformando quella prima evidenziata in blu nella seguente:

$$\begin{bmatrix} -\frac{5}{2} & \frac{9}{4} \\ 1 & 2 \end{bmatrix} \Longrightarrow \begin{bmatrix} -\frac{5}{2} & \frac{9}{4} \\ 0 & \frac{29}{10} \end{bmatrix}.$$

Anche per il termine noto, si potrà procedere come prima, ottenendo

$$\underline{b}^{(4)} = \begin{bmatrix} 0 \\ 1 \\ -\frac{1}{4} \\ 3 - \left(-\frac{2}{5}\right)\left(-\frac{1}{4}\right) \end{bmatrix}.$$

Sostituendo la sottomatrice 2×2 e il termine noto, è possibile ottenere quindi il seguente sistema lineare:

$$\underbrace{\begin{bmatrix} 2 & -1 & 1 & -2 \\ 0 & 2 & 0 & -1 \\ 0 & 0 & -\frac{5}{2} & \frac{9}{4} \\ 0 & 0 & 0 & \frac{29}{10} \end{bmatrix}}_{\underline{\underline{A}}^{(4)}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}}_{\underline{x}} = \underbrace{\begin{bmatrix} 0 \\ 1 \\ -\frac{1}{4} \\ \frac{29}{10} \end{bmatrix}}_{\underline{b}^{(4)}},$$

in cui è palese che $\underline{\underline{A}}^{(4)}$, che sarebbe il punto di partenza dell'ipotetico passo $k = 4$, sia una matrice triangolare superiore: abbiamo ottenuto il nostro obiettivo! Il passo $k = 4$ dunque non verrà effettuato, e noi considereremo $\underline{\underline{A}}^{(4)}$ e $\underline{b}^{(4)}$ come il nostro punto di arrivo. Per concludere, però, salviamo nella matrice $\underline{\underline{L}}^{(4)}$ anche l'ultimo coefficientino, ottenendo

$$\underline{\underline{L}}^{(4)} = \begin{bmatrix} * & * & * & * \\ 0 & * & * & * \\ \frac{1}{2} & \frac{1}{4} & * & * \\ 0 & 1 & -\frac{2}{5} & * \end{bmatrix}.$$

A questo punto, il sistema potrebbe essere risolto con la tecnica di sostituzione all'indietro, per scoprire che la soluzione in questo caso è $[1 \ 1 \ 1 \ 1]^{\mathrm{T}}$.

Cerchiamo ora di riassumere l'idea del metodo.

1. Prima di tutto, controlliamo che $a_{kk}^{(k)}$ sia diverso da 0; se questo elemento fosse nullo, scambiamo la k-esima riga con la prima successiva, sia questa la i-esima, avente $a_{ik}^{(k)} \neq 0$.
2. Calcolo m_{ik} per ogni i-esima riga, $i = k + 1, ..., n$

$$m_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}.$$

3. Rimpiazzo la i-esima riga, con $i = k + 1, ..., n$ con sé stessa, a cui sottraggo m_{ik} che moltiplica la k-esima riga. Faccio lo stesso per ciascuna i-esima componente del termine noto.
4. Proseguo fino a quando la matrice non diventa triangolare superiore; questo richiede $n - 1$ passi, dove n è il numero di righe e/o colonne della matrice $\underline{\underline{A}}$.

Questo algoritmo viene detto **metodo delle eliminazioni di Gauss** e, nel caso più generale ora mostrato, il suo costo computazionale è $n^3/3$. Infatti, un'implementazione di questo algoritmo, come si può intuire studiandolo, richiederebbe tre cicli annidati, e dunque sarebbe necessario un numero di operazioni aritmetiche elementari prossimo al cubo del numero delle righe della matrice.

Per chiarire alcuni punti, diversi testi, si veda per esempio [1, p. 127], riportano che il costo della fattorizzazione LU è circa pari a $2n^3/3$ anziché $n^3/3$. Questo testo si basa su [2, p. 45], che riporta esattamente il numero di operazioni richieste al fine di calcolare la fattorizzazione LU, e al termine di esse, indica $n^3/3$ come costo computazionale. Tutti i costi computazionali riportati su questo testo seguono questo testo come riferimento, e sono tra loro consistenti.

3.1 Alcune note su ciò che abbiamo appena fatto

Prima di tutto, osserviamo che, nonostante su di noi incombesse la minaccia di avere $a_{kk}^{(k)} = 0$ e quindi dover operare scambi di righe, questo non è successo. Esistono situazioni in cui si ha la certezza che questa cosa possa capitare; in particolare, questa certezza si ha quando:

- La matrice $\underline{\underline{A}}$ è a diagonale dominante per righe.
- La matrice $\underline{\underline{A}}$ è a diagonale dominante per colonne.
- La matrice $\underline{\underline{A}}$ è simmetrica definita positiva.

Si noti che la $\underline{\underline{A}}$ del nostro esempio non soddisfa alcune di queste condizioni, ma comunque non è stato necessario operare scambi: è stato un caso fortuito. Quando si ha una di queste condizioni si sa *a priori* che non serve operare scambi e quindi si ha la certezza che l'algoritmo è più semplice.

Un'altra osservazione: se la matrice $\underline{\underline{A}}$ è simmetrica, e non si effettuano scambi (dovuti a $a_{kk}^{(k)} = 0$), la sottomatrice sarà ancora simmetrica, **a ogni passo**; questo permette di calcolare meno elementi e, quindi, risparmiare conti: in particolare il costo delle eliminazioni di Gauss si dimezza (e si arriva così a $n^3/6$).

Un'osservazione conclusiva riguarda un metodo lievemente diverso di memorizzare gli elementi delle matrici. In particolare, è possibile, in alcuni codici, trovare come output una matrice nella forma

$$\begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ m_{21} & u_{22} & u_{23} & u_{24} \\ m_{31} & m_{32} & u_{33} & u_{34} \\ m_{41} & m_{42} & m_{43} & u_{44} \end{bmatrix} = \begin{bmatrix} 2 & -1 & 1 & -2 \\ 0 & 2 & 0 & -1 \\ \frac{1}{2} & \frac{1}{4} & -\frac{5}{2} & \frac{9}{4} \\ 0 & 1 & -\frac{2}{5} & \frac{29}{10} \end{bmatrix},$$

dove abbiamo mostrato a membro destro come si presenterebbe nel caso del nostro esempio. Al posto di memorizzare gli zeri, sono stati memorizzati i moltiplicatori, i coefficienti della combinazione lineare; in questo modo, nel caso di sistemi molto grossi, si può evitare di memorizzare due matrici e quindi risparmiare memoria.

3.2 Matrici triangolari e come ottenerle

Ci siamo presi la premura di salvare i coefficienti della combinazione lineare m_{ij}, e li abbiamo presi talmente in considerazione da porci il problema di memorizzarli in maniera efficiente. Ma perché? A cosa servono?

Consideriamo le seguenti due matrici.

$$\underline{\underline{L}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{4} & 1 & 0 \\ 0 & 1 & -\frac{2}{5} & 1 \end{bmatrix} \qquad \underline{\underline{U}} = \begin{bmatrix} 2 & -1 & 1 & -2 \\ 0 & 2 & 0 & -1 \\ 0 & 0 & -\frac{5}{2} & \frac{9}{4} \\ 0 & 0 & 0 & \frac{29}{10} \end{bmatrix}.$$

Da dove nascono queste matrici? Beh, $\underline{\underline{L}}$ è semplicemente la matrice contenente tutti i coefficienti delle combinazioni lineari, con, in più, tutti uni sulla diagonale. $\underline{\underline{U}}$, invece, non è altri che $\underline{\underline{A}}^{(4)}$, ovvero la matrice triangolare superiore ottenuta dalle eliminazioni di Gauss. In altre parole, tutti gli elementi di queste matrici non sono altro che risultati del metodo delle eliminazioni di Gauss. A questo punto, mi chiedo: quanto vale il loro prodotto, ovvero, $\underline{\underline{L}}\,\underline{\underline{U}}$? Un segmento di codice, e le sue uscite, valgono più di mille parole. A questo scopo, implementiamo, per l'esempio che abbiamo preso in considerazione, un codice che calcoli questo prodotto:

```
>> L = [1     0   0   0;
        0     1   0   0;
       1/2  1/4   1   0;
        0     1 -2/5   1];

>> U = [2  -1   1   -2;
        0    2   0   -1;
        0    0 -5/2  9/4;
        0    0   0   29/10];

>> L*U
ans =
    2.0000   -1.0000    1.0000   -2.0000
         0    2.0000         0   -1.0000
    1.0000         0   -2.0000    1.0000
         0    2.0000    1.0000    1.0000
>>
```

Cioè, abbiamo ritrovato la matrice $\underline{\underline{A}}$ di partenza! Cioè. Abbiamo scritto la matrice $\underline{\underline{A}}$ come il prodotto di due fattori: una matrice triangolare inferiore $\underline{\underline{L}}$, e una matrice triangolare superiore $\underline{\underline{U}}$. Dal momento che $\underline{\underline{L}}$ e $\underline{\underline{U}}$ dunque **fattorizzano** la matrice $\underline{\underline{A}}$, con grande fantasia, questa fattorizzazione è stata chiamata **fattorizzazione LU**. Le lettere L e U identificano *lower* e *upper*, come dire *triangolare inferiore* e *triangolare superiore*. Abbiamo capito, ora, a cosa serviva tenere in memoria i moltiplicatori. ;-)

Dopo questo esempio un po' noioso, inizia la parte davvero interessante dell'argomento, poiché ora potremo provare questa fattorizzazione per risolvere dei veri problemi, e iniziare a studiare altre fattorizzazioni valide in altre situazioni.

Riferimenti bibliografici

[1] A. Quarteroni e F. Saleri, "Introduzione al calcolo scientifico," 3ª edizione, Springer-Verlag Italia, Milano, 2006.

[2] G. Monegato, "Metodi e algoritmi per il calcolo numerico," *CLUT*, Torino, Settembre 2008.