

# **Manuale d'uso di FPM<sup>c</sup>**

*Poderico Luigi*

## **Introduzione**

Il presente documento fa parte della documentazione relativa al programma FPM<sup>c</sup>, nato dalla traduzione in c-ANSI di un programma scritto in Fortran presso il Dipartimento di Economia del Wye College (Università di Londra), nel 1967.

FPM<sup>c</sup> risolve, usando il metodo di Monte Carlo, due classi di problemi strettamente legati alla programmazione lineare, che verranno discusse nel paragrafo seguente.

L'uso di un metodo non deterministico per la risoluzione di un problema di quasi programmazione lineare, è dettato dalla necessità di ottenere più soluzioni ragionevolmente buone, allo scopo di condurre un'analisi comparativa con le soluzioni determinate in un ambiente di lavoro reale.

La fase di traduzione da Fortran a c-ANSI ha avuto tre scopi ben precisi:

1. rendere riusabile un programma che prevedeva l'input dei dati mediante schede perforate;
2. assicurare che il programma calcoli in modo corretto le soluzioni che determina;
3. usare le potenzialità del c per ottenere un programma che sia il più veloce possibile.

Mettiamo in guardia l'utente dal fatto che FPM<sup>c</sup>, così come il programma originario, non prevede alcun controllo sull'ammissibilità del problema in input. In questi casi si può incorrere in output insoliti.

## **Tipi di problemi risolubili**

FPM<sup>c</sup> è in grado di risolvere due classi di problemi di quasi programmazione lineare, così come mostrato nei sottoparagrafi seguenti.

### **Problema 1**

Consideriamo il problema classico della *programmazione lineare*, espresso nella formulazione matematica seguente:

$$\begin{array}{ll} \max & cx \\ & Ax \leq c \\ & \min \leq x \leq \max \\ & x \geq 0 \end{array} \quad \mathbf{PL}$$

dove

$c$  è il vettore dei ricavi;

$A$  è la matrice delle combinazioni;

$min$  e  $max$ , rispettivamente, il vettore delle quantità minime e massime producibili;

$x$  il vettore delle incognite.

Estendiamo **PL** con due classi di vincoli, eventualmente vuote:

### Vincoli di Preferenza

Associamo ad ogni bene prodotto un *fattore di preferenza*, cioè un numero compreso tra 0 e 100 che sta ad indicare quanto un prodotto è preferibile rispetto agli altri.

### Vincoli di Cardinalità

Il secondo vincolo che vogliamo imporre al nostro problema è quello di fissare il numero massimo di beni producibili rispetto a quelli potenzialmente tali. Più formalmente, di determinare il sottinsieme di attività, di cardinalità fissata, che massimizza il guadagno.

Come si vede, la classe di problemi appena esaminata è molto vasta ed adattabile a molte situazioni reali: questo fa di FPM<sup>c</sup> un valido strumento per l'analisi comparata delle pianificazioni adottate in ambienti lavorativi reali.

### **Problema 2**

La seconda classe di problemi risolti da FPM<sup>c</sup> rappresenta una variante al **Problema 1** quando, ad ogni unità di bene prodotto, viene associato un acquisto o una vendita aggiuntiva. Considerando **PL** (cioè trascurando i vincoli di preferenza e di cardinalità), la formulazione matematica del **Problema 2** è la seguente:

$$\begin{aligned} \max \quad & cx - be^+ + se^- \\ & Ax \leq c \\ & Dx = e \quad \quad \quad \mathbf{PL}' \\ & \min \leq x \leq \max \\ & x \geq 0 \end{aligned}$$

Dove:

$$e^+_i = \text{se } (e_i \geq 0, e_i, 0);$$

$$e^-_i = \text{se } (e_i \geq 0, 0, -e_i);$$

$b > 0$  è il vettore dei costi aggiuntivi;

$s > 0$  è il vettore dei guadagni aggiuntivi.

E' facile verificare come **PL'** sia riconducibile a **PL** e quindi **Problema 2** a **Problema 1**; risulta però utile l'utilizzo di **Problema 2** per studiare l'effetto di variazioni di costi, ovvero effettuare un'analisi del tipo *what-if* partendo dal problema principale.

## ***Il programma***

Il programma, al fine di mantenere un'alta portabilità, è stato scritto in c-ANSI. Questo garantisce, o almeno dovrebbe, l'uso del programma su macchine diverse, previa compilazione.

L'input e l'output dei dati avviene mediante file ascii, da compilare/visualizzare mediante un editor di testi o, preferibilmente, mediante un programma di input/output avanzato.

Essendo il Monte Carlo un metodo probabilistico, FPM<sup>C</sup> risulta molto lento per input già di media grandezza. Per questo motivo il numero massimo di attività programmabili è 64. Nulla vieta, in un prossimo futuro, di usare algoritmi "più deterministici" che possano far alzare questo limite.

Abbiamo già messo in guardia l'utente sul fatto che FPM<sup>C</sup> non ha nessun tipo di controllo sull'ammissibilità del problema in esame. Sommando a questo la caratteristica del c-ANSI di non possedere controlli run-time sull'overflow delle variabili e sugli indici dei vettori, non c'è da stupirsi se in output vi sono dei valori "strani".

Nel 90% dei casi si dovranno ricontrollare i dati di input per verificarne la correttezza. Se questo non bastasse per rientrare nella normalità, allora ci troveremo di fronte ad un errore di programmazione non determinato in fase di debugging, resa, quest'ultima, difficile dall'intrinseca indeterminatezza del metodo di Monte Carlo.

Evitiamo in questa sede di descrivere il sorgente di FPM<sup>C</sup>, che troverete come allegato all'intera documentazione, e che merita senz'altro uno studio approfondito, soprattutto la tecnica usata per implementare il Monte Carlo, merito indiscusso dei colleghi londinesi.

## ***L'interfaccia in Excel***

Come abbiamo già detto, FPM<sup>C</sup> prevede l'input e l'output mediante file ascii; l'utility, che speriamo si riveli utile per la compilazione di tali file, è stata redatta mediante l'uso di Excel 4.0. Il suo utilizzo non dovrebbe rivelarsi complesso, data la veste grafica e l'uso di macro associate a bottoni.

Per creare il file di input per FPM<sup>C</sup> basta partire dal foglio *Principale* e inserire i dati richiesti, spostandosi tra i vari fogli Excel mediante i bottoni *Avanti* e *Indietro*, fino ad arrivare all'ultimo foglio che prevede un bottone di *Esportazione*.

La pressione di questo bottone causa la creazione di un file ascii pronto per la lettura da parte di FPM<sup>C</sup>. La sua esecuzione dipende dal sistema operativo ospite; nel caso si usi l'MS-DOS, basterà digitare:

```
fpm file_di_input >file_di_output
```

Per leggere il file di output di FPM<sup>c</sup> basta cliccare il bottone *Importazione* dal foglio *Principale*.