

# AMPL Plus: Introduzione all'uso

A cura di Renato Bruni, Giovanni Fasano, Giampaolo Liuzzi, Sara Mattia

## Preliminari

In questa breve introduzione verranno chiariti alcuni aspetti di base relativi all'uso del Modellatore AMPL Plus, il quale costituisce un potente ambiente per la modellazione di problemi di Programmazione Matematica (lineare, nonlineare, intera). L'aspetto dell'interfaccia grafica che il Modellatore supporta è molto simile a quello delle applicazioni in ambiente Windows, consentendo così una maggior facilità d'uso.

Laddove l'utilizzo di alcune potenzialità di AMPL Plus non risultasse chiaro, si invita il lettore a consultare il manuale in linea (**Help** della barra dei menu). Vengono di seguito fornite alcune informazioni per l'installazione del software (Student Edition), che è possibile scaricare all'indirizzo: <ftp.cplex.com/pub/amplplus/setup.exe>

Il Software AMPL Plus può essere installato in ambiente Windows 95/98/NT/2000 ed è necessario che il PC sia dotato di processore 486DX o superiore; inoltre consente alcune potenzialità per le quali può rendersi necessaria la presenza di un coprocessore matematico (non indispensabile per Student Edition). Il software richiede almeno 16Mbyte di memoria RAM per funzionare correttamente.

- Per installare il software è necessario eseguire i seguenti passi:
  1. inserire il cd-rom (o il primo floppy disk) nel quale è presente il software;
  2. selezionare **Run** dal menu **Start** e digitare il comando **d:setup** (o analogo per floppy);
  3. premere **OK** e seguire i messaggi di istruzione che vengono visualizzati di volta in volta.
- Per disinstallare il software utilizzare la funzione **Add/Remove Programs** all'interno del **Control Panel**.

## Nozioni di base

Una volta installato AMPL Plus viene creata un'icona sul desktop di Windows: cliccando su tale icona viene aperta una finestra come nella seguente figura, che costituisce un'interfaccia (denominata GUI) per l'utente di AMPL Plus.

Nella finestra è presente un'area centrale destinata a contenere altre finestre, mentre nella parte superiore è presente una **Menu Bar** ed una **Tool Bar**, tipiche di programmi in ambiente Windows; infine alla base della finestra vi è la **Status Bar**. Si noti anche che all'interno della finestra sono presenti altre 3 finestre "minimizzate" denominate rispettivamente: **Commands**, **Solver Status** e **Model**. Le tre barre appena menzionate assolvono sinteticamente i seguenti compiti:

- **Menu Bar:** controlla in generale l'intera interfaccia grafica GUI (Graphic User Interface), presenta un layout tipico di Windows (cfr. seguente figura 2);
- **Tool Bar:** costituisce un modo rapido per accedere a comandi della Menu Bar di uso più frequente. È possibile far/non far comparire questa barra selezionando la relativa opzione nel menu **View** della Menu Bar (cfr. seguente figura 3);

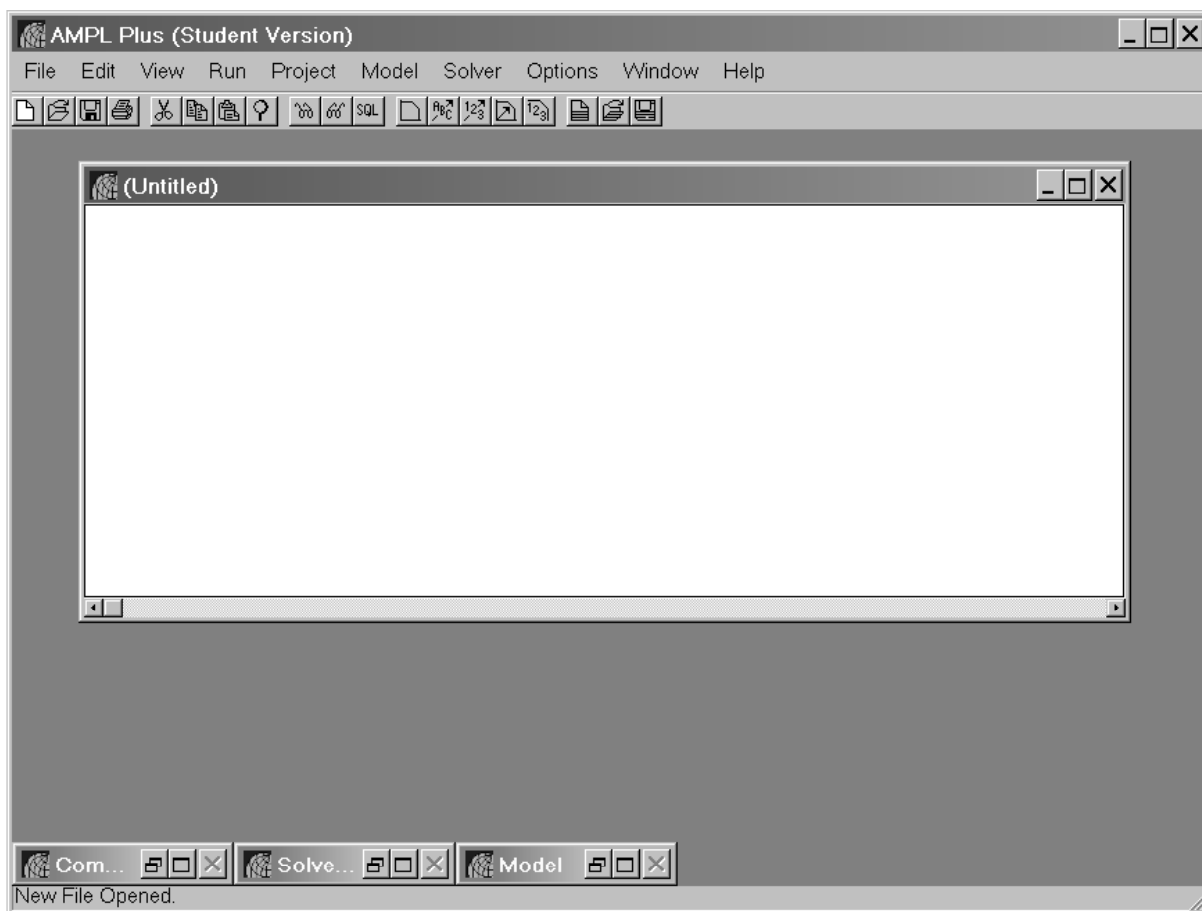


Figure 1:

- **Status Bar:** mostra all'utente messaggi relativi allo stato del modellatore, nonché informazioni associate all'ultima azione svolta dal modellatore sul problema scelto (cfr. seguente figura 4).

Per le tre finestre “minimizzate”, contenute nella GUI si ha quanto segue:

- **Commands:** è una finestra divisa in due settori. In quello superiore (area messaggi) vengono visualizzati in maniera sequenziale gli eventuali comandi inseriti dall'utente nonché i messaggi (errori od informazioni generati in maniera automatica), segnalati dal modellatore in fase di compilazione del modello, creato dall'utente. Nel settore inferiore è invece presente la riga di prompt del modellatore, segnalata dal simbolo **ampl:** (cfr. seguente figura 5):
- **Solver Status:** è una finestra nella quale vengono sintetizzate informazioni relative alla soluzione del problema sottoposto al modellatore, qui viene indicato il tipo di solutore impiegato (scelto dall'utente), il valore della funzione obiettivo ed un riassunto della struttura del problema. Da tale finestra è anche possibile interrompere *momentaneamente* (**Pause**) o *definitivamente* (**Stop**) la soluzione del problema, riprendendola eventualmente in un secondo tempo; in tal modo è possibile valutare lo stato di avanzamento della soluzione, senza attendere l'arresto dell'algoritmo (cfr. seguente figura 6).

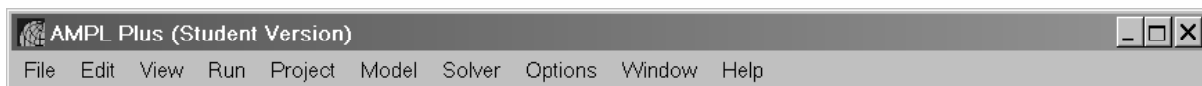


Figure 2:



Figure 3:

- **Model:** in tale finestra vengono descritte in maniera estesa caratteristiche ed oggetti definiti nel modello; premendo inoltre il tasto **Data** è possibile visualizzare il valore assunto all'ottimo dalle grandezze (variabili, parametri, etc.) presenti nel modello stesso (cfr. seguente figura 7).

## Modalità di utilizzo

L'utente che intende utilizzare l'interfaccia AMPL Plus per risolvere problemi lineari/nonlineari/interi, deve in primo luogo "tradurre" il problema fisico in uno equivalente che possa essere compreso dall'interfaccia. In questo caso il linguaggio per tradurre il problema originario è denominato **AMPL**. Per il momento ci limitiamo a dare una breve descrizione dei passi necessari all'utente può costruire e risolvere un generico modello, mediante l'interfaccia AMPL Plus. Per descrivere e risolvere un problema (che chiameremo per semplicità *pippo*) mediante il linguaggio AMPL e l'interfaccia AMPL Plus, occorre creare due file di testo denominati **pippo.mod** (l'estensione *.mod* è obbligatoria) e **pippo.dat** (anche qui l'estensione *.dat* è obbligatoria).

- il file **pippo.mod** contiene la dichiarazione dei parametri e delle variabili, la struttura e la descrizione dei vincoli e della funzione obiettivo, ovvero in sintesi contiene gli aspetti che costituiscono gli elementi fondamentali di un qualsiasi problema di programmazione matematica. In altri termini questo file contiene la descrizione vera e propria del problema di partenza; la sintassi con cui viene costruito questo file, come si è detto, è quella propria di AMPL;
- il file **pippo.dat** contiene invece i valori numerici che assumono i parametri del problema descritto in **pippo.mod**; questa diversificazione delle caratteristiche nei due file consente una notevole flessibilità all'interfaccia AMPL Plus. Infatti in tal modo risulta semplice risolvere lo stesso problema a partire da due istanze diverse, senza l'esigenza di doverlo ridescrivere completamente.

Al termine della creazione dei due file da parte dell'utente, è necessario farli leggere e compilare dal programma, ciò si ottiene percorrendo la sequenza di operazioni preliminari:

- scegliere l'opzione **Build Model** nel menu **Run** (o alternativamente premere **F7**) per leggere e compilare il file *.mod*. Al termine di questa fase AMPL Plus risponde aprendo la finestra **Solver Status** e comunicando all'utente lo stato di avanzamento nella compilazione del problema;
- scegliere l'opzione **Build Data** nel menu **Run** (o alternativamente premere **F8**) per leggere i dati del problema dal file *.dat* ed assegnarli a coefficienti e parametri presenti nel file *.mod*. Analogamente al caso precedente AMPL Plus risponde aprendo la finestra **Solver Status**.

A questo punto è possibile risolvere il problema (scegliendo l'opzione **Solve Problem** nel menu **Run** o alternativamente premendo **F9**) e successivamente leggere i risultati del processo di risoluzione, nella finestra **Model** e nelle finestre con label *xxxxx.qry* create scegliendo l'opzione **Data** nella finestra **Model**. Alternativamente, volendo ottenere i risultati finali in un'unica tabella, si può procedere scegliendo l'opzione **Build Results** (o **F10**) nel menu **Run**; conseguentemente AMPL Plus creerà un'unica finestra nella quale verranno inseriti i risultati finali.



Figure 4:

```

Commands
write bc:\PROGRA~1\AMPLSTED\dieti;
option outopt '';
option mip_priorities;
option mip_priorities ''; #not defined
CPLEX 6.5.3:
Root relaxation solution time = 0.00 sec.

      Nodes
      Node Left   Objective  IInf  Best Integer   Cuts/
      Node Left   Objective  IInf  Best Integer   Best Node  ItCnt   Gap
*      0+   0     118.0594    2     118.0594     118.1838    3
*      20+  5     120.6900    0     120.6900     118.8451   25   2.08%
*      28   4     119.7900    0     119.7900     118.9624   30   0.79%
*      28   4     119.3000    0     119.3000     118.9624   30   0.28%

solution C:\PROGRA~1\AMPLSTED\dieti.sol;
CPLEX 6.5.3: optimal integer solution; objective 119.3000000000003
34 MIP simplex iterations
33 branch-and-bound nodes

ampl: |

```

Figure 5:

Nella versione Student Edition sono presenti all'interno di AMPL Plus solo due solutori denominati **CPLEX** e **MINOS**: il primo può essere utilizzato per risolvere problemi lineari e interi, il secondo consente di risolvere in generale problemi nonlineari. È possibile scegliere da parte dell'utente il solutore da utilizzare od eventualmente aggiungere altri solutori; la scelta del solutore avviene tramite l'opzione **Solver** nella barra dei menu.

## I progetti

L'interfaccia AMPL Plus impone in generale una convenzione sulla denominazione dei file che gestisce, per esempio si è già avuto modo di osservare che il file del modello e quello relativo ai dati hanno rispettivamente l'estensione *.mod* e *.dat*: si noti che tale convenzione è vincolante nel senso che in caso di mancato rispetto, i comandi nel menu **Run** possono non funzionare correttamente. Il seguente prospetto riassume tali convenzioni, di cui comprenderemo appieno il significato tra breve:

Tipo di file	Estensione	Esempio
modello	<i>.mod</i>	esempio2.mod
dati	<i>.dat</i>	esempio2.dat
comandi	<i>.run</i>	dieti.run
interrogazione	<i>.qry</i>	make.qry
progetto	<i>.amp</i>	steelx.amp

Vediamo di chiarire l'uso delle utime tre tipologie di file. Finora per ottenere la soluzione del problema di partenza (denominato per esempio *pippo*), abbiamo avuto la necessità di introdurre una coppia di file *pippo.mod*

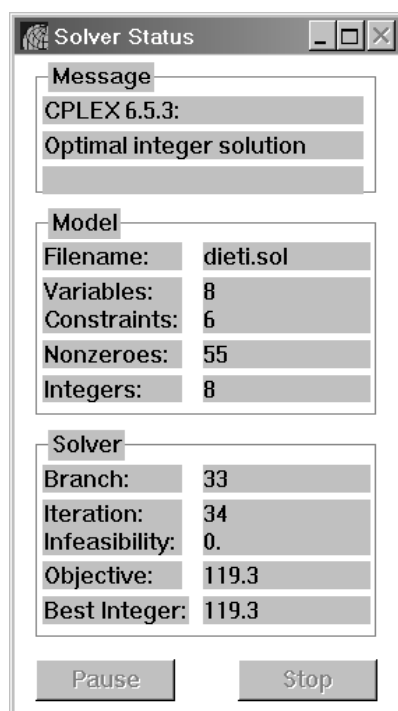


Figure 6:

e *pippo.dat* che ne rappresentassero le caratteristiche. Si supponga ora di voler rendere più flessibile tale schema, introducendo la possibilità di descrivere il problema di partenza mediante un numero di file maggiore di due. Per esempio potrebbe essere utile creare un solo file con estensione *.mod* associato al problema di partenza e poi studiare differenti scenari possibili, ognuno dei quali descritto da un file con estensione *.dat*. Di converso potremmo splittare il file *.mod* in più file con estensione *.mod*, con il fine di poter modificare agevolmente solo una parte delle specifiche del problema di partenza. Tutto ciò può essere realizzato con AMPL Plus introducendo i **Progetti**.

Un progetto è un insieme di file (con estensioni *.mod*, *.dat* e *.run*) che vogliono definire un problema anche complesso o più in generale uno scenario. Ciascuno di tali file (le cui specifiche non devono entrare in conflitto tra loro) contribuisce a definire una parte del problema o dello scenario; complessivamente l'insieme di tali file viene identificato a sua volta con un unico file di estensione *.amp* (cfr. Figura 8). Per esempio se i file *pippo1.mod*, *pippo2.mod*, *pippo.dat* e *pippo.run* formano un progetto, questo potrebbe essere identificato dal nome **pippo.amp**.

Un progetto può essere creato, aperto e modificato utilizzando le opzioni **New**, **Open** e **Edit** del menu **Project**, ma *non è possibile* aprire due progetti contemporaneamente. Chiariamo un ultimo dettaglio sulla struttura di un progetto; il file *.run* rappresenta un file di comandi creato dall'utente, il quale così può comunicare ad AMPL Plus esattamente le specifiche della soluzione che intende ottenere (scelta del solutore, stampa dei risultati, etc.).

Una volta aperto un progetto, per eseguirlo non è necessario seguire i passi visti nella sezione precedente, ma è sufficiente scegliere l'opzione **Solve Problem** nel menu **Run** (oppure premere **F9**); ciò sarà equivalente ad aver scelto in successione le tre opzioni **Build Model**, **Build Data** e **Solve Problem** del menu **Run**. Volendo poi visualizzare i valori assunti all'ottimo dalle grandezze definite nel problema, basterà aprire la finestra **Model** (normalmente "minimizzata") e selezionare al suo interno la grandezza richiesta; per la grandezza selezionata si aprirà un'ulteriore finestra, nella quale sarà possibile scegliere una pluralità di informazioni ad essa associate.

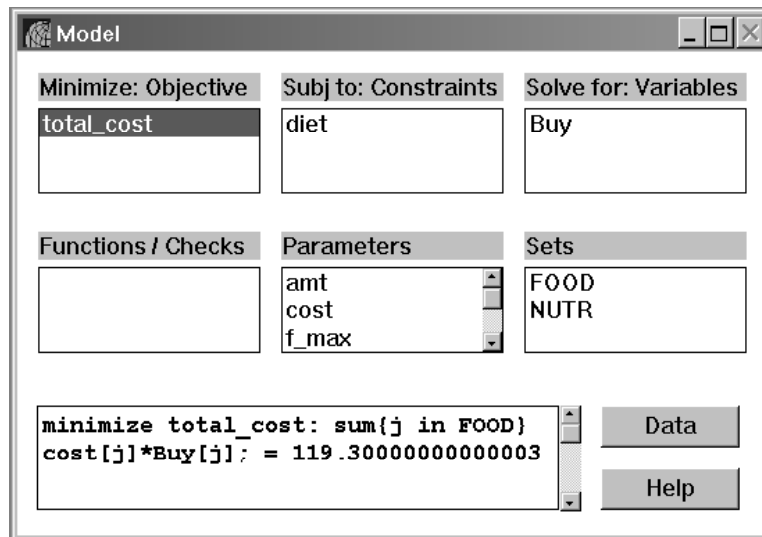


Figure 7:

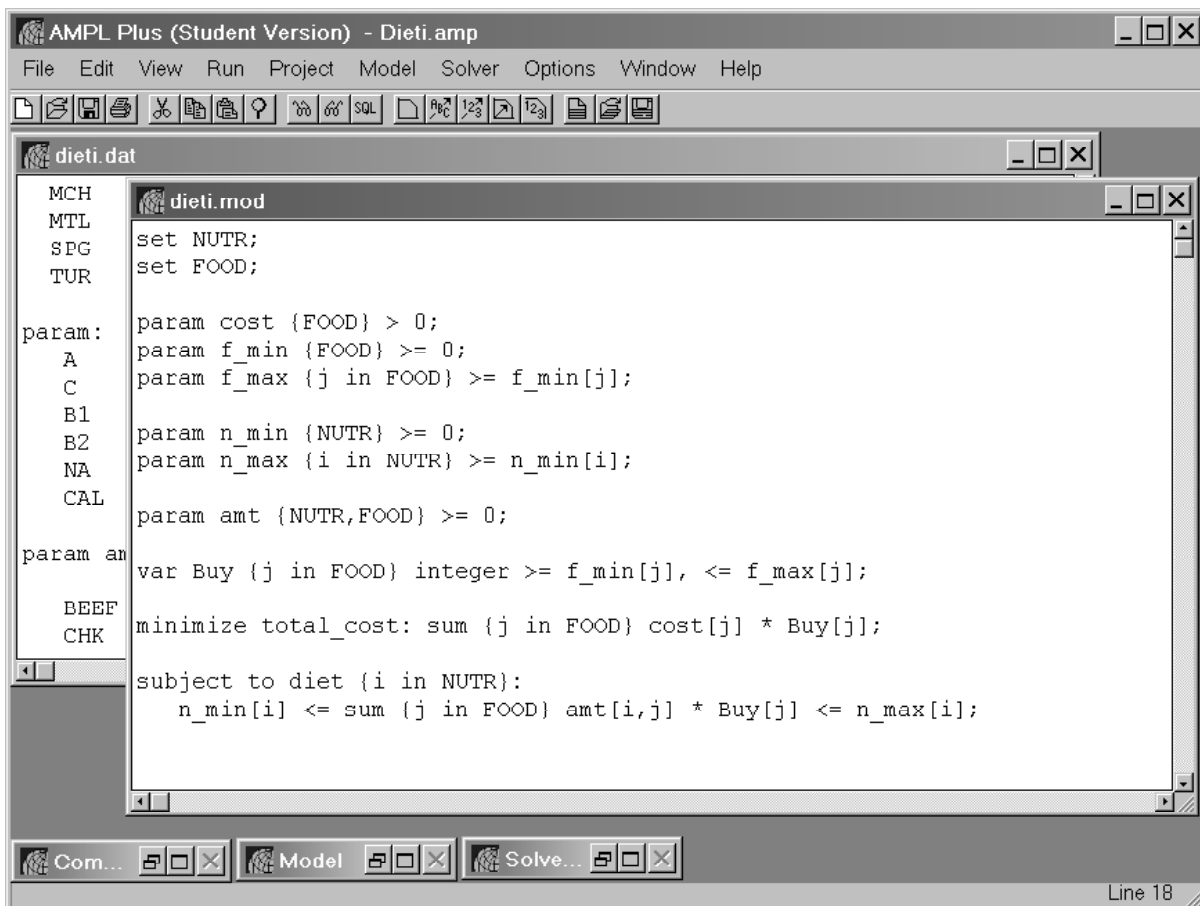


Figure 8: