Figure con gnuplot e xfig

Laboratorio Multimediale, a.a. 2003-2004

9 novembre 2003

1 Utilizzo di `xfig` per fare disegni geometrici

La figura 1 a pagina 2 è stata preparata col programma interattivo `xfig` utilizzando l'esportazione in formato *eps*. I comandi utilizzati nel documento LaTeX per inserire la figura sono i seguenti:

```
\begin{figure}
\begin{center}
\includegraphics[scale=0.6]{pitagora.eps}
\end{center}
\caption{Questa figura \'e stata fatta con \xfig.}
\label{fig:xfig1}
\end{figure}
```

Notiamo come i *font* utilizzati da `xfig` per scrivere le variabili a, b e c sono diversi da quelli utilizzati dal LaTeX. Inoltre col programma `xfig` non si possono scrivere tutte le formule che è possibile scrivere col LaTeX.

Per ovviare a questo problema possiamo utilizzare il *package* `psfrag`. La figura 2 è stata ottenuta col seguente codice LaTeX:

```
\begin{figure}
\begin{center}
\psfrag{a2+b2=c2}{$a^2+b^2=c^2$}
\psfrag{a}{$a$}
\psfrag{b}{$b$}
\psfrag{c}{$c$}
\includegraphics[scale=0.6]{pitagora.eps}
\end{center}
\caption{Questa figura \'e uguale alla figura~\ref{fig:xfig1} ma
il testo \'e stato rimpiazzato utilizzando \texttt{psfrag}.}
\label{fig:xfig2}
\end{figure}
```

In alternativa si sarebbe potuto ottenere lo stesso risultato utilizzando lo *special flag* del testo in `xfig` esportando quindi in formato `combined PS+LaTeX`.

2 Inserire grafici di funzioni fatti con `gnuplot`

Il programma `gnuplot` può essere utilizzato per disegnare grafici di funzioni in 2 o 3 variabili. Normalmente `gnuplot` visualizza i grafici su una finestra a video. Per mandare l'output di `gnuplot` su file bisogna utilizzare i comandi `set terminal` e `set output`.

Nella figura 3 si può vedere il grafico creato da `gnuplot` utilizzando come tipo di output il `terminal latex`. Questo metodo utilizza i comandi standard del LaTeX

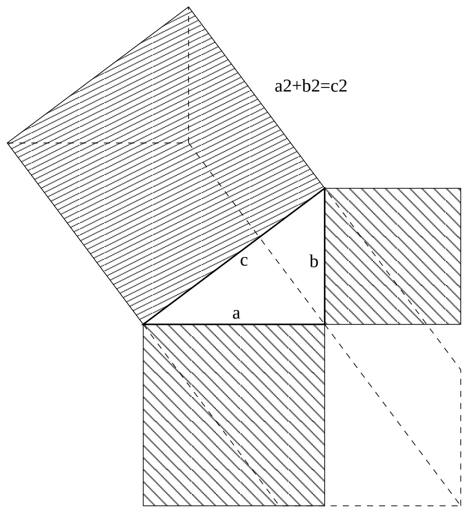

Figura 1: Questa figura è stata fatta con `xfig`.

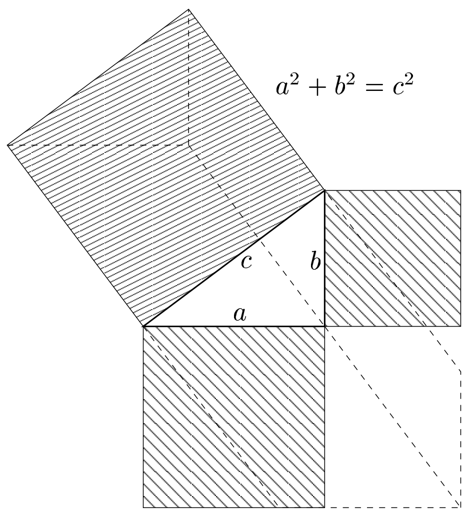

Figura 2: Questa figura è uguale alla figura 1 ma il testo è stato rimpiazzato utilizzando `psfrag`.

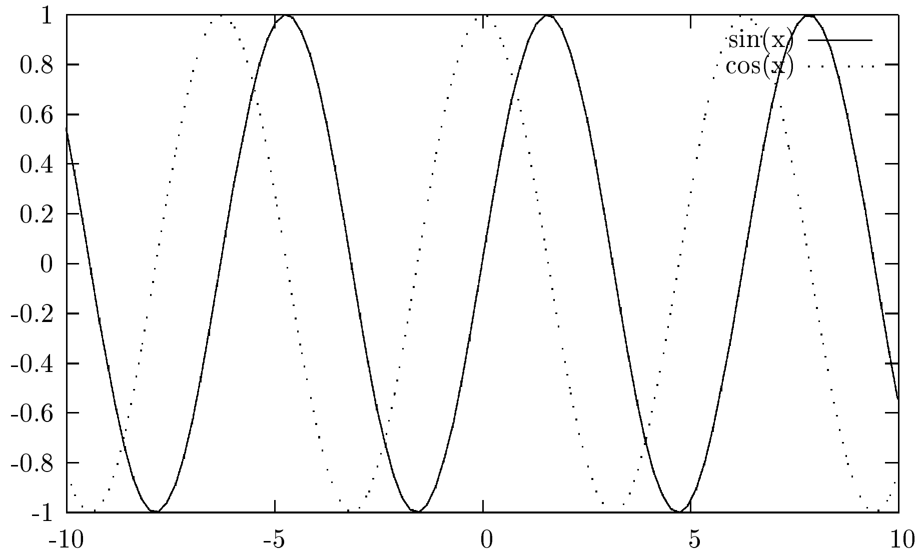

Figura 3: Questa figura è stata fatta con `gnuplot`, driver `latex`

per generare la figura e per questo non necessita neanche l'utilizzo del *package* `graphicx`. Questo metodo è però consigliabile solo per creare disegni molto semplici.

I comandi per generare la figura 3 con `gnuplot` sono i seguenti (vedi il *file* `figura3.gnuplot`):

```
set terminal latex
set output "sincos.tex"
plot sin(x), cos(x)
```

I comandi utilizzati per inserire la figura nel documento LaTeX sono invece i seguenti:

```
\begin{figure}
\input{sincos.tex}
\caption{Questa figura \'e stata fatta con \gnuplot, driver
\texttt{latex}}
\label{fig:sincos}
\end{figure}
```

Si noti come viene inserito il file LaTeX `sincos.tex` utilizzando semplicemente il comando `\input`.

Nella figura 4 vediamo una figura creata sempre da `gnuplot` ma utilizzando il `terminal epslatex`. Ecco i comandi dati a `gnuplot`:

```
set terminal epslatex
set output "gaussiana.eps"
set isosamples 30
set hidden
set ztics 0.5
a=3
set xrange[-a:a]
set yrange[-a:a]
set view 40,30
```

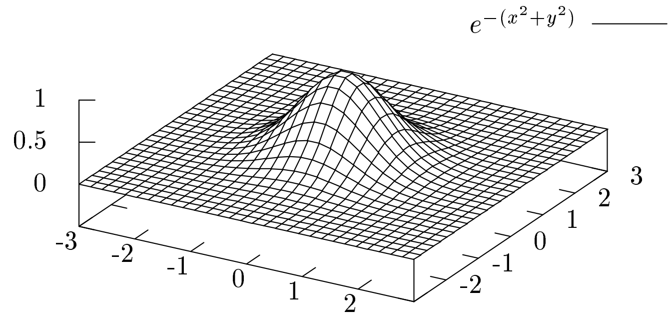

Figura 4: Questa figura è stata fatta con `gnuplot`, driver `epslatex`

```
set size 0.8,0.8
splot exp(-(x**2+y**2)) title "$e^{-(x^2+y^2)}$"
```

Con questo metodo `gnuplot` crea due diversi *files*: il file in formato *eps* (Encapsulated PostScript) che contiene il disegno e il file *tex* (in formato LaTeX) che contiene invece le scritte e le formule.

Ecco il codice LaTeX utilizzato per inserire la figura 4:

```
\begin{figure}
\begin{center}
\input{gaussiana.tex}
\end{center}
\caption{Questa figura \`e stata fatta con \gnuplot,
driver \texttt{epslatex}}
\label{fig:gaussiana}
\end{figure}
```

Si noti come anche in questo caso è sufficiente includere il file `gaussiana.tex` con il comando `input`; il file `gaussiana.eps` viene infatti inserito automaticamente tramite il comando `\includegraphics` che si trova nel file `gaussiana.tex`.

Nella figura 5 vediamo un'altro grafico generato da `gnuplot` utilizzando il `terminal eps`. I comandi dati a `gnuplot` sono i seguenti:

```
set terminal postscript eps
set output "toro.eps"
set isosamples 25
set hidden
set parametric
R=1.0
r=0.2
set urange [-pi:pi]
set vrange [-pi:pi]
splot (R+r*cos(u))*cos(v), (R+r*cos(u))*sin(v), r*sin(u) title "TITOLO"
```

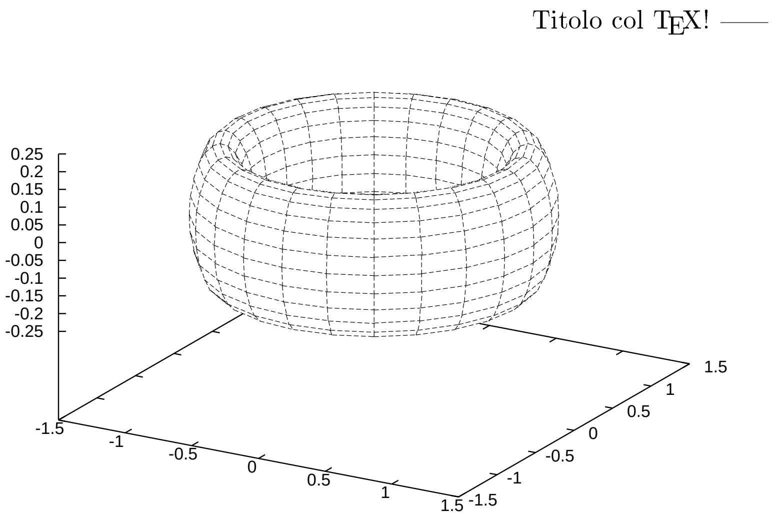

Figura 5: Questa figura è stata fatto con `gnuplot`, driver `postscript eps`. Il titolo è stato scritto utilizzando `psfrag`.

In questo caso `gnuplot` crea un unico file in formato *eps* che contiene sia la grafica che il testo. Abbiamo poi utilizzato il *package* `psfrag` del LaTeX per sostituire il titolo scritto da `gnuplot` con un titolo scritto in LaTeX.

Ecco il codice LaTeX utilizzato per inserire questa figura:

```
\begin{figure}
\psfrag{TITOLO}[r][r]{Titolo col \TeX!}
\includegraphics[scale=0.9]{toro.eps}
\caption{Questa figura \`e stata fatto con \gnuplot,
driver \texttt{postscript eps}.
Il titolo \`e stato scritto utilizzando \texttt{psfrag}.}
\label{fig:toro}
\end{figure}
```