

Due rivisitazioni

Riccardo Nisi

14 settembre 2008

Sommario

Considerazioni sulla gestione dei comandi ripetuti in un ciclo e il tracciamento di una curva in ambiente PSTricks con `\psplot` in coordinate parametriche e polari.

1 Introduzione

Il pacchetto `\multido`, se utilizzato in un documento insieme a numerosi altri pacchetti, può non riconoscere le proprie variabili e bloccare la compilazione. Per aver sperimentato l'inconveniente ho voluto risolvere la ripetizione di un comando che agisce su valori che cambiano con altri due cicli compatibili con \LaTeX . Il `loop ... repeat` del codice \TeX e il `n{ ... } repeat` del `PostScript`

Un altro approccio abbastanza elaborato è stato quello con alcuni grafici dell'ambiente `PSTricks`, e con il codice `PostScript` che descrive le equazioni delle curve. Così ho pensato che la mia esperienza potesse essere condivisa con chi come me trova stimolante cercare di capire (e interpretare), pur empiricamente e a piccoli passi, situazioni che possono apparire intricate e fuori portata.

Indice

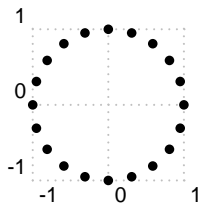
1	Introduzione	1
2	Comandi ripetuti	2
2.1	Circonferenza punteggiata	2
2.2	Spirale archimedeana	3
3	Ellissi e iperboli	4
3.1	Ellisse in coordinate parametriche	5
3.2	Ellisse in coordinate polari	6
3.3	Iperbole in coordinate polari	6
4	Riferimenti bibliografici	9

2 Tracciamenti con comandi ripetuti.¹

Negli esempi i comandi ripetuti sono `\psdot` e `\ndot`, il primo disegna una circonferenza di punti, il secondo una spirale tracciata dal carattere A. Entrambi utilizzano le coordinate polari ed effettuano rotazioni, come nel caso di una lancetta d'orologio, incrementando l'ampiezza dell'angolo della specifica coordinata polare e, nel secondo caso, anche la lunghezza del raggio. `\psdot` di default disegna un punto nell'estremo mobile del raggio. Ma agendo nell'ambiente opzioni -con `dotstyle=square, triangle, ...` - si possono rappresentare altri oggetti. Un po' diverso è `\nput` che può essere pensato come un'asta rigida di lunghezza variabile: una estremità è legata al nodo, l'estremità libera si connette rigidamente al centro del carattere A; la base di A è parallela alla retta dell'asta.

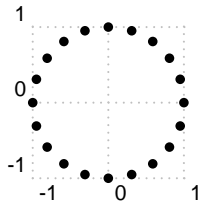
2.1 Circonferenza punteggiata

- Circonferenza punteggiata e il suo codice: il comando `\multido`.



```
\begin{pspicture}[showgrid=true](-1,-1)(1,1)
  \multido{\iAmp=18+18}{20}{\psdot(1;\iAmp)}
\end{pspicture}
```

- Circonferenza punteggiata e il codice T_EX: il ciclo `\loop ... \repeat`. L'ultimo punto, nella posizione 360, è tracciato quando il contatore assume il valore 342.



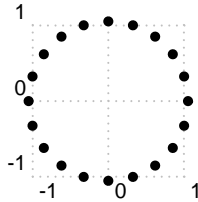
```
\begin{pspicture}[showgrid=true](-1,-1)(1,1.2)
  \newcount\cont \cont=0
  \loop
    \ifnum\cont<343 \advance \cont by 18 \psdot(1;\the\cont)
  \repeat
\end{pspicture}
```

Meno convenzionale è il caso del codice PostScript. Qui il punto, disegnato come un cerchio pieno di raggio 1.5 pt, è rappresentato 20 volte nella stessa posizione (30,0). Il comando per disegnare la circonferenza è `arc` e i suoi cinque argomenti sono x_c y_c r ang_i ang_f . Il comando `rotate` di PostScript agisce sugli assi cartesiani. Sono loro a ruotare

¹Le soluzioni con il comando `multido` sono gli esempi 5-14-1 e 6-2-54 di *LaTeX Graphics Companion, Second Edition*

di 18° e facilitano le cose, almeno numericamente. Certo se vivessimo sull'asse x vedremmo disegnare un solo punto.

■ Circonferenza punteggiata e il codice PostScript.



```
\begin{pspicture}[showgrid=true](-1,-1)(1,1)
\pscustom{
\code{
30 0 moveto
20{30 0 1.5 0 360 arc
gsave
fill
grestore
18 rotate
stroke} repeat
}}
\end{pspicture}
```

2.2 Spirale archimedeana

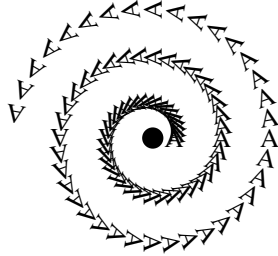
Nel caso della spirale archimedeana disegnata con il carattere A il punto di riferimento è un nodo tracciato come un cerchio pieno da `\cnode` che ne stabilisce le coordinate del centro, lunghezza del raggio e nome. Sia `\multido` che il `loop ... repeat` di \TeX spostano, ruotano insieme al raggio e tracciano il carattere A disponendolo secondo una spirale archimedeana con il comando `\nput` (coordinate polari), che prende come riferimento il nodo, e che assegna la direzione che varia di una quantità costante ad ogni passo del ciclo e la distanza dal centro a cui porre l'etichetta, vista come `\labelsep`, crescente con uno step costante.

`\multido`, che gestisce due variabili indipendenti, traccia facilmente la spirale. In questo esempio la macro `\multido` esegue 90 volte il comando `\nput[opzioni] {ang. rot.} {nome nodo rifer.}{oggetto-etich.}`. `rot` va da 0° a 900° , due rotazioni complete più metà. Le due assegnazioni sono la rotazione, `rot=\nA`, e `labelsep=\rB pt`, la distanza che separa nodo ed etichetta. La prima variabile di `\multido`, `\nA`, accetta solo numeri interi, la seconda variabile `\rB` accetta numeri reali; le due variabili sono indipendenti, e avendo entrambe un valore costante anche il loro rapporto lo è. Ed è sul valore 20 di tale rapporto che ci si basa per tracciare con il codice \TeX una seconda spirale uguale alla prima.

Meno lineare è il compito di `loop ... repeat` che ha una sola vera variabile, intera. `\cont` è un contatore adeguato a rappresentare i valori interi degli angoli, ma non può rappresentare la lunghezza del raggio, che ha valori razionali. Per questo motivo il valore intero di `\cont` è passato sotto forma di distanza alla variabile razionale `\dimen2`, che può essere divisa. Ponendo anche il passo di `\cont=10` e dividendo `\dimen2` per 20 si ottiene una spirale uguale alla precedente.

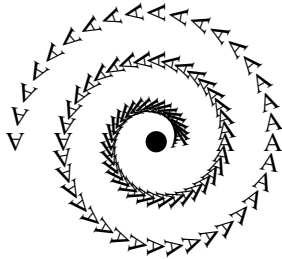
La distanza costante tra due spire è 18pt (0.64cm).

- Spirale archimedeana e `\multido`.



```
\begin{pspicture}(2,3.5)
\cnode*(2.5,1){4pt}{A}
\multido{\nA=0+10,
         \rB=0+0.5}{90}{%}
  \nput[rot=\nA,%
        labelsep=\rB pt]{\nA}{A}{A}}
\end{pspicture}
```

- Spirale archimedeana e il codice `TEX`.



```
\begin{pspicture}(2,2.5)
\newcount\cont \cont=0
\psset{unit = .4cm}
\cnode*(6,0){4pt}{A}
\loop
  \ifnum\cont<900 \advance \cont by 10
    \dimen2=\cont pt \divide\dimen2 by 20
  \nput[rot=\the\cont,
        labelsep=\dimen2]{\the\cont}{A}{A}
\repeat
\end{pspicture}
```

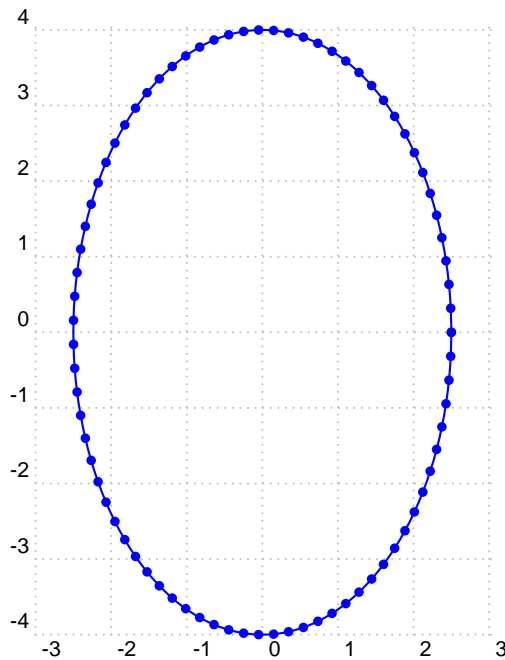
3 Ellissi e iperboli

`PSTricks` traccia una curva in coordinate parametriche con la macro `\parametricplot` e in coordinate polari con `\psplot` esplicitando in questo caso la scelta con l'assegnazione `polarplot=true` in `\psset` o in `\psplot`. Le coordinate parametriche e l'equazione del

raggio polare dell'ellisse sono ricavati dalla sua equazione cartesiana riferita all'origine del sistema di riferimento. Le coordinate parametriche sono, $(a \cos t, b \sin t)$, l'equazione polare del raggio è $r = \frac{ab}{\sqrt{b^2 \cos^2 t + a^2 \sin^2 t}}$ (t è l'angolo tra il raggio vettore e l'asse x ; $\frac{x}{a} = \cos t$, $\frac{y}{b} = \sin t \Rightarrow \frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$). Il radicando è positivo $\forall t$, la curva, se $a > b$, esiste ed è continua in $|x| \leq a$ (se $a < b \Rightarrow |x| \leq b$) e non ha complicazioni di tracciamento. In `\psplot` sia le coordinate polari che l'equazione del raggio sono espresse in codice `PostScript` assegnando negli esempi ad `a` e `b` nel caso delle coordinate parametriche 2.5 e 4 e il contrario in quello del raggio polare. L'esempio del tracciamento dell'ellisse è utile per il confronto con quello più problematico della iperbole.

Interpretando il codice `PostScript` del raggio dell'ellisse, scritto secondo la Reversed Polish Notation (omologa al modo in cui i dati del `PostScript` sono salvati in uno stack disposti in una pila in cui l'ultimo elemento salvato è il primo ad essere estratto), il primo operatore a destra è `div`, che agisce su un dividendo e un divisore. Si scorre a sinistra per individuare il divisore e si incontra l'operatore `sqrt` che ha un solo operando. Per individuarlo si scorre a sinistra e si incontra l'operatore `add`, che agisce su due addendi. Occorre individuarli scorrendo sempre a sinistra: c'è un `mul` che a sua volta agisce su due fattori; il primo è `a dup mul` (a^2 andando da destra a sinistra), il secondo fattore è `x sin dup mul` ossia $\sin^2 x$. Considerando anche il primo `mul` incontrato, si ottiene il primo addendo che è $a^2 \sin^2 x$. Analogamente si individua il secondo addendo che è $b^2 \cos^2 x$. La somma dei due addendi trovati è il radicando, somma di due termini positivi. La sua radice quadrata è il divisore che si stava cercando. Scorrendo l'espressione ancora a sinistra è facile individuare il dividendo in `a b mul`, cioè ab .

3.1 Ellisse in coordinate parametriche



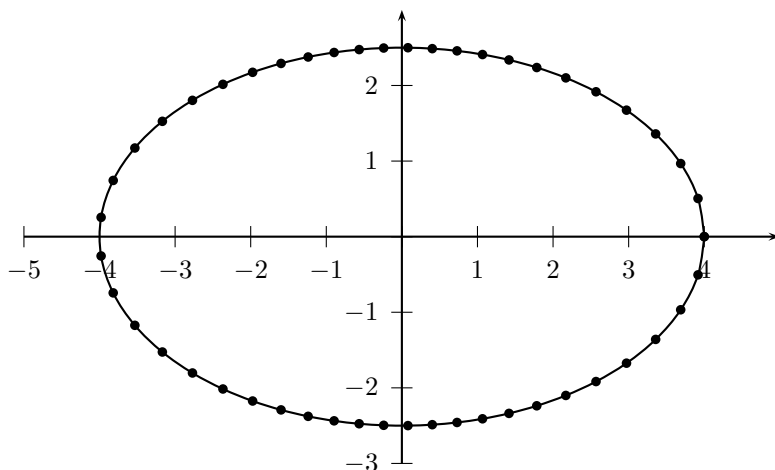
```
Codice ellisse in coordinate parametriche
\def\Ellissey{t cos 2.5 mul t sin 4 mul} % variabile angolare in gradi
\def\Ellissex{t RadtoDeg cos 4 mul t RadtoDeg sin 2.5 mul} % var. ang. in rad
```

```

\begin{pspicture}[showgrid=true](-3,-4)(3,4)
\psset{linestyle=solid, unit=1cm}
\parametricplot[plotstyle=curve,plotpoints=80,showpoints=true,
linecolor=blue]{0}{360}{\Ellissey}
\end{pspicture}

```

3.2 Ellisse in coordinate polari



```

Codice ellisse in coordinate polari
\resetOptions
\begin{pspicture}(-4,-1)(4,2)
\psaxes[ticks=all]{->}(0,0)(-5,-3)(5,3)
\psset{linestyle=solid, unit=1cm,polarplot=true}
\psplot[plotstyle=curve, showpoints=true]{0}{360}
{/a 4 def /b 2.5 def a b mul x cos dup mul b dup mul mul
x sin dup mul a dup mul mul add sqrt div}
\end{pspicture}

```

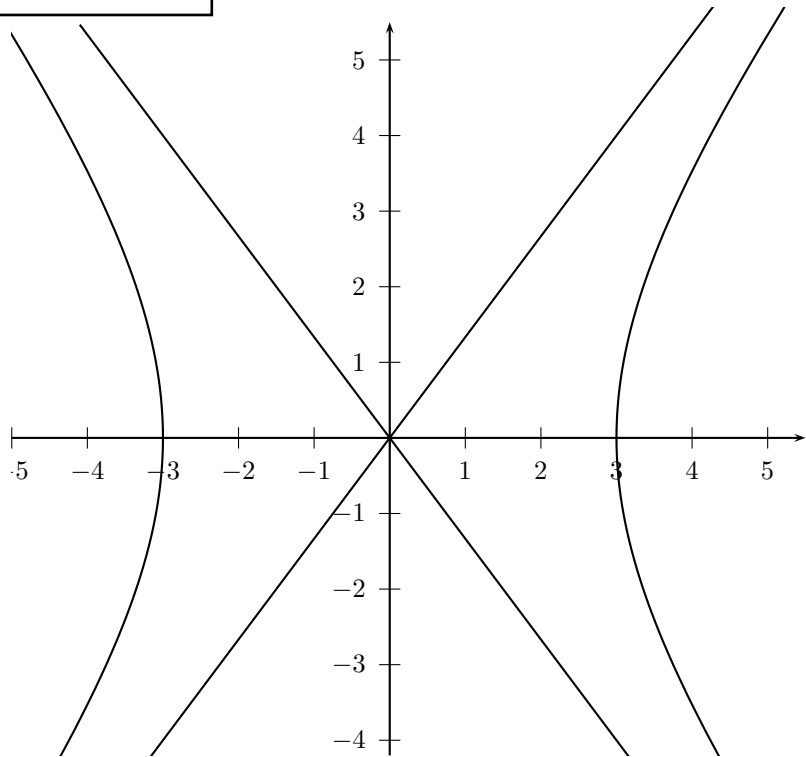
3.3 Iperbole in coordinate polari

L'equazione del raggio polare dell'iperbole è $r = \frac{ab}{\sqrt{b^2 \cos^2 t - a^2 \sin^2 t}}$ e poiché il radican-
do è una differenza la curva non è continua, esiste per $|x| \geq a$ e t interno agli angoli tra
gli asintoti contenenti l'asse x , e il suo tracciamento richiede attenzione. Infatti la macro
`\psplot` che la traccia deve, in un certo senso, essere guidata dall'utente. Se si prova a far
calcolare il raggio in $0 \leq t \leq 360$, come per l'ellisse, o in intervalli non congrui si creano
incompatibilità che bloccano la compilazione. Occorre inoltre tracciare i due rami di iperbole
separatamente assegnando due intervalli per la variabile t basati sulle direzioni degli asintoti.
Per determinare tali direzioni senza calcolatrice ho usato un po' di codice PostScript immes-
so nel documento \LaTeX con la macro `\pstverb` di `PSTricks`. La `v` minuscola di garantisce
che l'output di PostScript sia indipendente da quello di \LaTeX , tanto da poterlo sovrascrivere
(`\vglue1.8cm` ha lo scopo di separare gli output dei due codici). Il codice PostScript cal-
cola gli angoli tra gli asintoti e l'asse x e li stampa in un box accanto all'iperbole permettendo
di scegliere gli intervalli per `\psplot`: più gli estremi degli intervalli sono vicini agli angoli

fra gli asintoti e l'asse x più i rami di iperbole saranno lunghi. Poiché gli estremi degli archi nel cui intervallo calcolare il raggio polare e tracciare la curva sono determinati esternamente è opportuno che `\psplot` effettui un controllo di positività sul radicando.

`\psplot` là dove aspetta gli angoli non accetta il codice `PostScript`, anche se calcolare con `atan` gli angoli in tale ambiente sembrerebbe il modo più diretto per farlo.

Asintoti iperbole:
 angolo I Q = 53.1301
 angolo II Q = 126.87
 angolo III Q = 233.13
 angolo IV Q = -53.1301



Codice PostScript per calcolare gli angoli tra asintoti e asse x e tracciare il box con i risultati introdotto con la macro `\pstverb` di `PSTricks`.

```
\pstverb
{
/LM -20 def           %pone il margine sinistro a -20pt
/Times-Roman findfont 10 scalefont setfont
/nstr 8 string def %crea una stringa vuota di 8 caratteri
/a 3 def
/b 4 def
/arctg1
  { b a atan }def      %calcola l'arctg b/a I Q
```

```

/arctg2
  { b a neg atan }def          %calcola l'arctg b/a II Q
/arctg4
  { b neg a atan 360 sub }def  %calcola l'arctg b/a IV Q
/arctg3
  { b neg a neg atan }def     %calcola l'arctg b/a III Q
/prt-n
  {nstr cvs show}def         %converte il primo numero dello stack in una str di 8 caratteri
/prtArctg1
  { ( angolo I Q = ) show
    arctg1 prt-n }def        % prepara una stringa col valore dell'arctg}def
/prtArctg2
  { ( angolo II Q = ) show
    arctg2 prt-n }def
/prtArctg3
  { ( angolo IIIQ = ) show
    arctg3 prt-n }def
/prtArctg4
  { ( angolo IVQ = ) show
    arctg4 prt-n }def
LM 15 moveto
( Asintoti iperbole:) show
LM 0 moveto
prtArctg1                    %stampa l'arctg1 nella posizione indicata
LM -15 moveto
prtArctg2
LM -30 moveto
prtArctg3
LM -45 moveto
prtArctg4
newpath                      % traccia il box dei risultati
-40 30 moveto
-40 -55 lineto
90 -55 lineto
90 30 lineto
-40 30 lineto
1 setlinewidth
stroke showpage
}

```

Codice iperbole in coordinate polari

```

\begin{pspicture*}(-5,-4.2)(5.5,5.7)
\psaxes[ticks=all]{->}(0,0)(-5,-4.2)(5.5,5.5)
\psset{linestyle=solid, unit=1cm,polarplot=true}

```

```

%Tracciamento condizionato iperbole I e IV Q
\begin{pspicture*}(-5,-4.2)(5.5,5.7)
\psaxes[ticks=all]{->}(0,0)(-5,-4.2)(5.5,5.5)

```



```

\psset{linestyle=solid, unit=1cm,polarplot=true}
\psplot[plotstyle=curve, showpoints=false]{-50}{50}
{/a 3 def /c 5 def /b{c dup mul a dup mul sub sqrt} def
/rdcn{b dup mul x cos dup mul mul a dup mul x sin dup mul mul sub}def
rdcn 0 gt{a b mul rdcn sqrt div} if %se rdcn>0 -> istruzioni nelle graffe
}
%Tracciamento condizionato iperbole II e III Q
\psplot[plotstyle=curve, showpoints=false]{127}{230}
{/a 3 def /c 5 def /b{c dup mul a dup mul sub sqrt} def
/rdcn{b dup mul x cos dup mul mul a dup mul x sin dup mul mul sub}def
rdcn 0 gt{a b mul rdcn sqrt div}{} ifelse %se rdcn>0 -> istruz I parentesi else II
}
%Asintoti
\psset{linestyle=solid, unit=1cm,polarplot=false}
\psplot[plotpoints=30, showpoints=false]{-4}{4.5}
{/a 3 def /c 5 def /b{c dup mul a dup mul sub sqrt} def
b a div x mul}
\psplot[plotpoints=30, showpoints=false]{-4.1}{3.5}
{/a 3 def /c 5 def /b{c dup mul a dup mul sub sqrt} def
b a div x mul neg}
\end{pspicture*}

```

4 Riferimenti bibliografici

- [1] Claudio Beccari. *LaTeX Guida a un sistema di editoria elettronica*. HOEPLI, Milano, IT, 1991. ISBN 88-203-1931-4.
- [2] Dominique Rodriquez e Herbert Voß. additional macros for pstricks v. 3 05, 2008. URL <http://www.ctan.org/tex-archive/graphics/pstricks/contrib/pstricks-add/>.
- [3] Michel Goossens, Frank Mittelbach, Sebastian Rahtz, Denis Roegel e Herbert Voß. *The LaTeX Graphics Companion Second Edition*. Addison-Wesley, Reading, MA, USA, 2007.
- [4] Frank Mittelbach e Michel Goossens. *The LaTeX Companion Second Edition*. Addison-Wesley, Reading, MA, USA, 2007. ISBN 0201-136296. With Johannes Braams, David Carlisle e Chris Rowley.
- [5] Gianni Gilardi. *Il TeX Introduzione al linguaggio e complementi avanzati*. Zanichelli/Decibel, Bologna, IT, 1993.
- [6] Adobe System Inc. *PostScript Language, Tutorial and Cookbook*. Addison-Wesley, Reading, MA, USA, 1994. ISBN 0-201-10179-3.
- [7] Donald E. Knuth. *The TeXbook*. Addison-Wesley, Reading, MA, USA, 1986. ISBN 0-201-13447-0.
- [8] Leslie Lamport. *LaTeX: A Document Preparation System*. Addison-Wesley, Reading, MA, USA, 1985. ISBN 0-201-15790-X.
- [9] Massimiliano Dominici. Bibliografia con LaTeX, 2008. URL <http://www.guit.sssup.it/latex/bibliografia.php>.

- [10] Cambridge University. Bibliography with \LaTeX , 2008. URL
<http://www.eng.cam.ac.uk/help/tpl/textprocessing/bibliographies.html>.