

plotterart

Modul maker pro L^AT_EXovou sazbu esteticky produktivní grafiky

Vojtěch Žáček

11. listopadu 2005

Abstrakt

Tento dokument popisuje implementaci makra `plotterart`.
Program od verze 2.0 používá pro vykreslování a manipulaci s grafickými prvky makro `mfpic`.

Program je součástí bakalářské práce: obor aplikovaná informatika, Fakulta Informatiky, Masaríkova Universita.

1 `plotterart.tex`

`\plotterart` Příkaz zahájí `plotterart` grafiku. První parametr je název grafiky, druhý parametr určuje základní šířku grafiky v bodech. Třetím parametrem je druh použitého pera a čtvrtý parametr je základním rozměrem pera (viz. `\plainipen`).

```
1 < *tex >
2 \def\plotterart #1 #2 #3 #4 {
3   \message{*** BEGIN: plotterart #1 #2 #3 #4}
4   \opengraphsfile{#1}
5
6   \mfpic[1]{0}{#2}{0}{#2}
7   \mfsrc{
8     pla_name := #1;
9     pla_width := #2;
10    pla_pen := #3;
11    pla_fat := #4;
12  }
13  \plainifunc
14  \plainiglobal
15  \plainipoints
16  \plainitrans
17  \plainipen
18  \plainisym
19  \plaperspective
20 }
```

`\endplotterart` Makro ukončuje definici `plotterart` grafiky. V místě použití tohoto makra je grafika vykreslena.

```
21 \def\endplotterart {
```

```

22 \message{*** END: endplotterart}
23 \endmfpic
24 \closegraphsfile
25 }

```

\fan Použití: \fan X10 Y10 X20 Y20 X30 Y30 N

Makro vykresluje soustavu úseček. Parametry makra definují tři body:

$A = (X_{10}, Y_{10}), B = (X_{20}, Y_{20}), C = (X_{30}, Y_{30})$. Poslední parametr určuje počet vykreslených úseček. Makro vykreslí úsečky $A-D_i$. První bod úsečky je vždy zadaný bod A , druhý bod je prvkem množiny $\{B, \dots, D_{1/i}, \dots, C\}$, která vznikne rovnoměrným dělením úsečky $B-C$. Index i nabývá celých hodnot z intervalu $< 0, N >$.

```

26 \def\fan #1 #2 #3 #4 #5 #6 #7 {
27 \mfsrc{
28 x10 := plagetpointx(#1);
29 y10 := plagetpointy(#2);
30 x20 := plagetpointx(#3);
31 y20 := plagetpointy(#4);
32 x30 := plagetpointx(#5);
33 y30 := plagetpointy(#6);
34 for i = 0 step (1 / #7) until 1:
35 path pxx;
36 pxx =
37 z10
38 --
39 i[z20, z30];
40 plotterartdrawall(pxx);
41 endfor;
42 }
43 \hvhloc;
44 }

```

\join Použití: \join X11 Y11 X12 Y12 X21 Y21 X22 Y22 N

Makro vykresluje soustavu úseček. Parametry makra definují čtyři body:

$A = (X_{11}, Y_{11}), B = (X_{12}, Y_{12}), C = (X_{21}, Y_{21}), D = (X_{22}, Y_{22})$. Poslední parametr určuje *krok*. Makro vykreslí úsečky $E_i - F_i$. První bod leží na spojnici bodů A a B s dělením N . Druhý bod leží na spojnici bodů C a D s dělením N . (viz. \fan)

```

45 \def\join #1 #2 #3 #4 #5 #6 #7 #8 #9 {
46 \mfsrc{
47 x11 := plagetpointx(#1);
48 y11 := plagetpointy(#2);
49 x12 := plagetpointx(#3);
50 y12 := plagetpointy(#4);
51 x21 := plagetpointx(#5);
52 y21 := plagetpointy(#6);
53 x22 := plagetpointx(#7);
54 y22 := plagetpointy(#8);
55 for i = 0 step (1 / #9) until 1:
56 path pxx;
57 pxx =
58 i[(x11, y11), (x12, y12)]
59 --

```

```

60         i[(x21, y21), (x22, y22)];
61         plotterartdrawall(pxx);
62     endfor;
63 }
64 \hvloc;
65 }

\bfan    Použití: \bfan X10 Y10 X20 Y20 X30 Y30 N
\bfan je modifikací makra \fan. Nevykresluje soustavu úseček, ale Béziérových
křivek.

66 \def\bfan #1 #2 #3 #4 #5 #6 #7 {
67     \mfsrc{
68         x10 := plagetpointx(#1);
69         y10 := plagetpointy(#2);
70         x20 := plagetpointx(#3);
71         y20 := plagetpointy(#4);
72         x30 := plagetpointx(#5);
73         y30 := plagetpointy(#6);
74         for c=0 step (1 / #7) until 1:
75             path pxx;
76             pxx =
77                 z10{curl pla_curl}
78                 ..(pla_xbezier, pla_ybezier)..
79                 c[z20, z30]{curl pla_curl};
80             plotterartdrawall(pxx);
81         endfor;
82     }
83     \hvloc;
84 }

\bjoin    Použití: \bjoin X11 Y11 X12 Y12 X21 Y21 X22 Y22 N
\bjoin je modifikací makra \join. Nevykresluje soustavu úseček, ale Béziérových
křivek. Tvar křivky lze ovlivnit makry: \placurl \plabezierpoint.

85 \def\bjoin #1 #2 #3 #4 #5 #6 #7 #8 #9 {
86     \mfsrc{
87         x11 := plagetpointx(#1);
88         y11 := plagetpointy(#2);
89         x12 := plagetpointx(#3);
90         y12 := plagetpointy(#4);
91         x21 := plagetpointx(#5);
92         y21 := plagetpointy(#6);
93         x22 := plagetpointx(#7);
94         y22 := plagetpointy(#8);
95         for c = 0 step (1 / #9) until 1:
96             path pxx;
97             pxx =
98                 c[(x11, y11), (x12, y12)]{curl pla_curl}
99                 ..(pla_xbezier, pla_ybezier)..
100                c[(x21, y21), (x22, y22)]{curl pla_curl};
101             plotterartdrawall(pxx);
102         endfor;
103     }
104     \hvloc;
105 }

```

```

\playaxis      Vykreslí osu  $y$ .
106 \def\playaxis{
107   \mfsrc{
108     draw(pla_half_width, 0)--(pla_half_width, pla_width);
109   }
110 }

\plaxis      Vykreslí osu  $x$ .
111 \def\plaxis{
112   \mfsrc{
113     draw(0, pla_half_width)--(pla_width, pla_half_width);
114   }
115 }

\plabox      Vykreslí základní hranice grafiky. Čtverec o straně  $pla\_width$ .
116 \def\plabox {
117   \mfsrc{
118     draw pplabox;
119   }
120 }

\vsym      Zapne globální symetrii ve vertikální rovině.
121 \def\vsym{
122   \mfsrc{
123     glob_h := false;
124     glob_v := true;
125   }
126 }

\hsym      Zapne globální symetrii v horizontální rovině.
127 \def\hsym{
128   \mfsrc{
129     glob_h := true;
130     glob_v := false;
131   }
132 }

\nosym      Vypne globální symetrii v obou rovinách.
133 \def\nosym{
134   \mfsrc{
135     glob_h := false;
136     glob_v := false;
137   }
138 }

\hvsym      Zapne globální symetrii v obou rovinách.
139 \def\hvsym{
140   \mfsrc{
141     glob_h := true;
142     glob_v := true;
143   }
144 }

```

```

\vlloc      Zapne lokální symetrii ve vertikální rovině.

145 \def\vlloc{
146   \mfsrc{
147     loc_h := false;
148     loc_v := true;
149   }
150 }

\hloc      Zapne lokální symetrii v horizontální rovině.

151 \def\hloc{
152   \mfsrc{
153     loc_h := true;
154     loc_v := false;
155   }
156 }

\noloc     Vypne lokální symetrii v obou rovinách.

157 \def\noloc{
158   \mfsrc{
159     loc_h := false;
160     loc_v := false;
161   }
162 }

\hvloc     Zapne lokální symetrii v obou rovinách.

163 \def\hvloc{
164   \mfsrc{
165     loc_h := true;
166     loc_v := true;
167   }
168 }

\platransgroup  Nastaví transformační skupinu. 0 - defaultní transformační skupina (zrcadlení
                  kolem os  $X$  a  $Y$ ) 1 - otáčení kolem středového bodu  $z_{99}$ 

169 \def\platransgroup #1 {
170   \mfsrc{
171     tg := #1 * 4;
172   }
173 }

\placurl     Zakřivení Bézierovy křivky. zakřivení:
              > 1 větší, = 1 standardní, < 1 menší.

174 \def\placurl #1 {
175   \mfsrc{
176     pla_curl := #1;
177   }
178 }

\plarotated  Nastavení rotace všech následujících prvků o úhel  $\delta$ . Defaultní hodnota je 0.

179 \def\plarotated #1 {
180   \mfsrc{
181     delta := #1;
182   }
183 }

```

`\plascaled` Zvětšení v obou osách o hodnotu Ξ . Defaultní hodnota je 1. Při hodnotě > 1 dochází ke zvětšení zvětšení, a při hodnotě < 1 ke zmenšení.

```
184 \def\plascaled #1 {
185   \mfsrc{
186     chi := #1;
187     if pla_perspective:
188       pla_fat := pla_fat * chi;
189       show pla_fat;
190     else:%
191     fi;
192   }
193   \plainipen
194 }
```

`\plabezierpoint` Řídící bod Bézierovi křivky z_{42} . Defaultně je nastaven na výchozí hodnotu středového bodu z_{99} .

```
195 \def\plabezierpoint #1 #2 {
196   \mfsrc{
197     pla_xbezier := #1 * pla_width;
198     pla_ybezier := #2 * pla_width;
199   }
200 }
```

`\plafat` Nastavení tloušťky pera.

```
201 \def\plafat #1 {
202   \mfsrc{
203     pla_fat := #1 * chi;
204   }
205   \plainipen
206 }
```

`\plarand` Nastavi nahodne promenne TODO ... dopsat dokumentaci `\plarand`

```
207 \def\plarand #1 #2 #3 #4 #5 #6 #7 {
208   \mfsrc{
209     rand_x := #1;
210     rand_y := #2;
211     rand_real_x := #3;
212     rand_real_y := #4;
213     rand_delta := #5;
214     rand_fat := #6;
215     rand_chi := #7;
216   }
217 }
```

`\plaperspective` Zapne perspektivu. Zvětšení (`\plascale`) ovlivní i tloušťku pera.

```
218 \def\plaperspective{
219   \mfsrc{
220     pla_perspective := true;
221   }
222 }
```

`\planoperspective` Vypne perspektivu.

```
223 \def\planoperspective{
224   \mfsrc{
```

```

225     pla_perspective := false;
226   }
227 }

\plainiglobal   Inicialize globalních proměnných.

228 \def\plainiglobal{
229   \mfsrc{
230     boolean symetrie;
231     boolean glob_h;
232     boolean glob_v;
233     boolean loc_h;
234     boolean loc_v;
235     boolean pla_perspective;
236     half = .5;
237     pla_half_width = half * pla_width;
238     tg = 0;
239     pla_curl = 1;
240     delta = 0;
241     chi = 1;
242     pla_xbezier = pla_width * half;
243     pla_ybezier = pla_width * half;
244     pla_xcenter = pla_width * half;
245     pla_ycenter = pla_width * half;
246     rand_x = 0;
247     rand_y = 0;
248     rand_real_x = 0;
249     rand_real_y = 0;
250     rand_delta = 0;
251     rand_fat = 0;
252     rand_chi = 0;
253     boolean pla_cut;
254     pla_cut := false;
255   }
256 }

\plainipoints   Inicializace základních a řídicích bodů.

257 \def\plainipoints{
258   \mfsrc{
259     z1 = (0, 0);
260     z2 = (0, pla_width);
261     z3 = (pla_width, pla_width);
262     z4 = (pla_width, 0);
263     z5 = (pla_width * half, pla_width * half);
264     z11 = (0, pla_width * half);
265     z21 = (pla_width * half, pla_width);
266     z31 = (pla_width, pla_width * half);
267     z41 = (pla_width * half, 0);
268     z99 = (pla_xcenter, pla_ycenter);
269     z42 = (pla_xbezier, pla_ybezier);
270     path pplabox;
271     pplabox = z1--z2--z3--z4--cycle;
272   }
273 }

\plainitrans   Inicializace pole transformací.

```

```

274 \def\plainitrans{
275   \mfsrc{
276     transform t[];
277     t0 = identity;
278     t1 = identity reflectedabout (z11,z21);
279     t2 = identity reflectedabout (z31,z41);
280     t3 = identity reflectedabout (z11,z21)
281         reflectedabout (z31,z41);
282     t4 = identity;
283     t5 = identity rotatedaround (z99, 90);
284     t6 = identity rotatedaround (z99, 180);
285     t7 = identity rotatedaround (z99, 270);
286   }
287 }

\plainipen      Inicializace pole per.

288 \def\plainipen{
289   \mfsrc{
290     pen PERA[];
291     PERA1 = pensquare scaled pla_fat;
292     PERA2 = makepen(fullcircle scaled pla_fat);
293     PERA3 = PERA1 transformed
294         identity rotated 45;
295     PERA4 = PERA3 transformed
296         identity rotated 10;
297     if ((pla_pen > 0) and (pla_pen < 5)):
298       pickup PERA[pla_pen];
299     else:%
300       pickup defaultpen;
301     fi;
302   }
303 }

\plainifunc      Inicializace základních funkcí.

304 \def\plainifunc{
305   \mfsrc{
306     def plotterartdraw(expr pa)(expr tr)(expr out) =
307       if out:
308         t97 := identity rotatedaround (z99, plagetranddelta(delta));
309         t98 := identity scaled (plagetrandchi(chi));
310         t99 := identity shifted
311             (pla_half_width - (chi * pla_half_width)
312              + uniformdeviate rand_real_x,
313             pla_half_width - (chi * pla_half_width)
314              + uniformdeviate rand_real_y);
315       if pla_cut:
316         draw pa
317           transformed tr
318           transformed t97
319           transformed t98
320           transformed t99
321           cutbefore pplabox cutafter pplabox;
322       else:%
323         draw pa

```

```

324         transformed tr
325         transformed t97
326         transformed t98
327         transformed t99;
328     fi;
329     else:%
330     fi;
331     enddef;
332 }
333 \mfsrc{
334     def plotterartdrawall(expr pxxall) =
335         plotterartdraw(pxxall, t[tg+0], true);
336         plotterartdraw(pxxall, t[tg+1], (glob_h and loc_h));
337         plotterartdraw(pxxall, t[tg+2], (glob_v and loc_v));
338         plotterartdraw(pxxall, t[tg+3], (glob_h and loc_h
339                                     and glob_v and loc_v));
340     enddef;
341 }
342 \mfsrc{
343     vardef plagetpointx(expr val)
344         = ((val + (uniformdeviate rand_x)) * pla_width)
345     enddef;
346     vardef plagetpointy(expr val)
347         = ((val + (uniformdeviate rand_y)) * pla_width)
348     enddef;
349     vardef plagetrandidelta(expr val)
350         = (val + (uniformdeviate rand_delta))
351     enddef;
352     vardef plagetrandidchi(expr val)
353         = (val + (uniformdeviate rand_chi))
354     enddef;
355 }
356 }

\placuttrue

357 \def\placuttrue{
358     \mfsrc{
359         pla_cut := true;
360     }
361 }

\placutfalse

362 \def\placutfalse{
363     \mfsrc{
364         pla_cut := false;
365     }
366 }

\plainitrans      Inicializace symetrie. Globalni a lokalni symetrie jsou zapnuty.

367 \def\placclip{
368     \mfsrc{
369         clip currentpicture to pplabox;
370     }
371 }

```

`\plainitrans` Inicializace symetrie. Globalni a lokalni symetrie jsou zapnuty.

```
372 \def\plainisym{
373   \hvsym
374   \hvloc
375 }
376 \</tex>
```

2 plotterart.sty

```
377 \<*/sty>
378 \input plotterart.tex\relax
379 \</sty>
```