Let me start with a few notes about how this document with the pretty-printed MetaPost source code was created. Since John Hobby’s MetaPost is almost (but not quite) similar to Don Knuth’s METAFONT, it’s easy to use the MFT utility which converts METAFONT source to \TeX source.

Prior to this conversion it is necessary to apply a few modifications. This is best done by Ulrik Vieth’s SED script mp2mft.sed from CTAN:graphics/metapost/contrib/misc/. MFT should be invoked with the \-s mp2 option, also supplied by Ulrik’s bundle. The resulting file can be processed by either plain \TeX or (as for this application) by PDF\TeX.

In the first case it is necessary to include Tom Rokicki’s epsf.tex macros, in the latter you’ll need Hans Hagen’s supp-pdf.tex and the usual \pdfoutput=1 and \pdfcompresslevel=9.

Including MetaPost graphics in La\TeX documents is straightforward with the graphicx bundle and the latest pdftex.def by David Carlisle, Sebastian Rahtz, and Hans Hagen.

beginfig(1); % Start the first graphic.
% Declare a bunch of variables; points . . .
pair zast, leftrange, rightrange, bottomrange, toprange;
pair topleft, topright, bottomleft, bottomright;
path restriction_two, inrange; % paths . . .
picture cutlabel; % and an image.
% Set the coordinates; origin equals (0, 0).
  zast = origin; leftrange = (−4cm, 0); rightrange = (4cm, 0);
  bottomrange = (0, −1cm); toprange = (0, 4cm);
  topleft = (−4cm, 4cm); bottomleft = (−4cm, 0);
  topright = (4cm, 4cm); bottomright = (4cm, 0);
% MetaPost does not (yet) provide the possibility for plotting functions,
% so we have to use intermediate points for the parabola.
  restriction_two = (−2cm, 4cm)
    for i = −8 upto 8: . . . (i/4, (i/4) * (i/4)) * 1cm endfor;
% buildcycle creates a closed path from several (sub)paths.
inrange = buildcycle(bottomleft -- topleft -- (−2cm, 4cm),
  restriction_two, (2cm, 4cm) -- topright -- bottomright -- bottomleft);
% Note that PostScript works in an “additive” way, so we have to start
% with the background. Instead of ‘white’ you can use brilliant
% \TeXnicolor.
  fill inrange withcolor .8white;
% Introduce the x and y axis.
drawarrow leftrange -- rightrange;
drawarrow bottomrange -- toprange;
draw restriction_two; % Draw the parabola.
% Mark various points of interest. This is done by \TeX itself.
label lrt(btex ... etex, zast); % \TeX label: "$x^\ast$"
label bot(btex ... etex, rightrange); % \TeX label: "$x_1$"
label lft(btex ... etex, toprange); % \TeX label: "$x_2$"
cutlabel := thelabel rt(btex ... etex, zast + (0, 1cm)); % \TeX label: "$\nabla g_1(x^{\ast})$"
unfill bbox cutlabel; draw cutlabel;
label lft(btex ... etex, zast − (0, 1cm)); % \TeX label: "$\nabla g_2(x^{\ast})$"
\begin{verbatim}
cutlabel := thelabel top(btex ... etex, (3cm, 0));  % TeX label: "$g_1(x)=0$"
unfill bbox cutlabel; draw cutlabel;

cutlabel := thelabel lrt(btex ... etex, (2cm, 4cm));  % TeX label: "$g_2(x)=0$"
unfill bbox cutlabel; draw cutlabel;

cutlabel := thelabel(btex ... etex, (-3cm, 2cm));  % TeX label: "$P$"
unfill bbox cutlabel; draw cutlabel;

% Draw the two gradients with a somewhat thicker line.
pickup pencircle scaled 1.2pt;

drawarrow zast -- (zast + (0, 1cm));
drawarrow zast -- (zast - (0, 1cm));

endfig; % The first graphic is finished.
\end{verbatim}

The final output can be included into your TeX document in the usual fashion. If you want to create standard output you can use \texttt{\epsfbox{nlpgraph.1}}. For PDF output use \texttt{\convertMPtoPDF{nlpgraph.1}{1}{1}}.

LaTeX users should be familiar with what they have to do. Note that you \textit{must} rename the files written by MetaPost to something like \texttt{nlpgraph1.mps} for \texttt{pdftex.def}'s sake.

\begin{figure}
\centering
\includegraphics[width=0.5\textwidth]{nlpgraph1}
\end{figure}

OK, just as I wrote in my announcement in the \texttt{pdftex} list, the code for these graphics is rather quick and dirty. Without comments even I don’t quite understand what the following does. However, I just don’t have the time to fill the gaps now. Also the code could be improved by using more parameters and letting MetaPost do the calculations.

On the other hand, the main idea should be clear: define some fixed points and their relative positions, draw lines and fill areas, finally put in labels and special markers. And remember not to overwrite earlier stuff that should be visible.
\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure.png}
\caption{Graphical representation of the system.}
\end{figure}

\begin{verbatim}
beginfig(2);

pair rightrange, leftrange, toprange, bottomrange;

rightrange = (8cm, 0); leftrange = origin;
toprange = (0, 5cm); bottomrange = (0, \(-1cm\));

label bot(btex ... etex, rightrange); \% \TeX label: \"$x_1$\"
label lft(btex ... etex, toprange); \% \TeX label: \"$x_2$\"

pair xast; xast = (4cm, 2cm);

label llft(btex ... etex, xast); \% \TeX label: \"$\hat{x}_{\ast}$\"

path T; T = (1cm, 5cm) -- (7cm, \(-1cm\));
label lft(btex ... etex, point 0 of T); \% \TeX label: \"$T$\"
label(btex ... etex, (1cm, 2cm)); \% \TeX label: \"$g_{1}(x)=0$\"

path g_i; g_i = (.75toprange .. {(1, \(-1\)})xast .. .rightrange;

path L; L = (2cm, 5cm) .. {(1, \(-1\)}) .. {right}(8cm, 2cm);

pair SP, SPT, TSPT;

SP = L intersectionpoint ((xpart(xast), \(-\infty\)) -- (xpart(xast), \(\infty\)));
SPT = point .666667 of L;

label urt(btex ... etex, SP); \% \TeX label: \"$L(\hat{x}_{\ast}, u_{\ast})$\"
label urt(btex ... etex, SPT); \% \TeX label: \"$L(x, u_{\ast})$\"

TSPT = T intersectionpoint (SPT -- (xpart(SPT), \(-\infty\)));

path Konvex;
Konvex = buildcycle(L, SPT -- TSPT -- point 0 of T -- point 0 of L);

fill Konvex withcolor .8white;

for i = T, g_i, L: draw i; endfor;

draw xast -- SP; draw SPT -- TSPT;

drawarrow leftrange -- rightrange;
drawarrow bottomrange -- toprange;

endfig; \% 2
\end{verbatim}
beginfig(3);

pair rightrange, toprange;
rightrange = (8cm, 0); toprange = (0, 6cm);

for i = rightrange, toprange: drawarrow origin -- i; endfor;

label bot(btex ... etex, rightrange); % \TeX label: "$x_1$"
label lft(btex ... etex, toprange); % \TeX label: "$x_2$"

pair xast; xast = (5.5 xpart(rightrange), 3 cm);

path g[];
\draw g[1] = (8 xpart(rightrange), ypart(toprange)) {dir 200}
\draw g[2] = (9 xpart(rightrange), 4 ypart(toprange))
\draw g[3] = rightrange {dir 160} \draw g[3] = rightrange {dir 190};

path P; P = buildcycle(g[1], g[2], g[3]);
fill P withcolor 0.8 white;

for i = 1 upto 3: draw g[i]; endfor;

label rt(btex ... etex, point 0 of g[1]); % \TeX label: "$g_1(x)=0$"
label rt(btex ... etex, point 0 of g[2]); % \TeX label: "$g_2(x)=0$"
label urt(btex ... etex, point 0 of g[3]); % \TeX label: "$g_3(x)=0$"

numeric Txast;
(Txast, whatever) = g[2] intersection times
((xpart(xast), -infinity) -- (xpart(xast), infinity));

pair A, B, C;
A = - direction Txast of g[2];
B = xast + whatever * A; xpart(B) = xpart(rightrange);
C = xast + whatever * A; xpart(C) = 0;

draw C -- B dashed evenly; label rt(btex ... etex, B); % \TeX label: "$T$"

pair Left[], Middle[], Right[];
for i = -4 upto 2:
\draw Left[i] = (0cm, 5.5cm + 2i cm);
\draw Middle[i] = xast + i * (.5cm, .3cm);
\draw Right[i] = (xpart(rightrange), 5cm + 1i cm);
\draw Left[i] .. Middle[i]{A} .. Right[i] dashed evenly;
endfor;

label rt(btex ... etex, Right[0]); % \TeX label: "$f(x)=c$"

pickup pencircle scaled 1.2pt;

pair nabla, nablag;
\draw nabla = xast + (.7A rotated -90);
\draw nablag = xast + (.35A rotated -90);

picture cutlabel;
cutlabel := thelabel lft(btex ... etex, nabla); % \TeX label: "$\nabla f(x^{\ast})$"

unfill bbox cutlabel; draw cutlabel;
\cutlabel := \thelabel \lft (\text \ldots \text \rtex \text \nabla g_2(x^\ast)) \text \% \TeX \ label: "$\nabla g_2(x^\ast)$"
\unfill \bbox \cutlabel; \text \draw \cutlabel;
\drawarrow \xast \text \ldots \nabla f\text \ldots \nabla g_2\text \ldots \nabla g_3\text \ldots \nabla g_3$
\cutlabel := \thelabel \urt (\text \ldots \text \rtex, (1\text \ cm, 2.5\text \ cm)) \text \% \TeX \ label: "$P$"
\unfill \bbox \cutlabel; \text \draw \cutlabel;
\endfig \% 3
\xfig
beginfig(4);

pair rightrange, toprange;
rightrange = (8cm, 0); toprange = (0, 5cm);

label bot btex \ldots etex, rightrange; \hfill \text{\TeX\ label: } "x_1$"
label lft btex \ldots etex, toprange; \hfill \text{\TeX\ label: } "x_2$"

path g[], f;
g[1] = .3toprange{dir 25} ..
{up}{.5 xpart(rightrange), ypart(toprange)};
g[2] = .7toprange{right} .. {dir 250}.9rightrange;

label rt btex \ldots etex, point 1 of g[1]; \hfill \text{\TeX\ label: } "$g_1(x)=0$"
label urt btex \ldots etex, point 0 of g[2]; \hfill \text{\TeX\ label: } "$g_2(x)=0$"

pair xast;
xast = g[1] intersectionpoint g[2];

label top btex \ldots etex, xast; \hfill \text{\TeX\ label: } "$x^{\ast}$"

f = .5toprange .. xast .. (.8 xpart(rightrange), .9 ypart(toprange));

label rt btex \ldots etex, point 2 of f; \hfill \text{\TeX\ label: } "$f(x)$"

numeric Txast[];
pair A[], B[], C[];

(Txast[1], whatever) = g[1]intersectionpointtimes
((xpart(xast), -infinity) -- (xpart(xast), infinity));
C[1] = xast + whatever * (A[1] rotated 90); ypart(C[1]) = 0;

(Txast[2], whatever) = g[2]intersectionpointtimes
((xpart(xast), -infinity) -- (xpart(xast), infinity));

(Txast[3], whatever) = fintersectionpointtimes
((xpart(xast), -infinity) -- (xpart(xast), infinity));

path P;
P = buildcycle(origin -- C[1] -- xast, g[1], origin -- toprange);
fill P withcolor .9white;
P := buildcycle(C[2] -- rightrange, g[2], xast -- C[2]);
fill P withcolor .9white;

picture cutlabel;
cutlabel = thelabel lrt btex \ldots etex, B[3]; \hfill \text{\TeX\ label: } "$\nabla f(x^{\ast})$"
unfill bbox cutlabel; draw cutlabel;
cutlabel := thelabel rt btex \ldots etex, B[1]; \hfill \text{\TeX\ label: } "$\nabla g_{-1}(x^{\ast})$"
unfill bbox cutlabel; draw cutlabel;


\begin{verbatim}
cutlabel := thelabel lft(btex ... etex, B[2]);  \hfill \text{T\TeX\ label: "$\nabla g_2(x^\ast)$"}
unfill bbox cutlabel; draw cutlabel;

\hfill \text{T\TeX\ label: "$P$"}

for \(i = 1\) upto \(2\): draw \(x^\ast - - C[i]\) dashed evenly; endfor;
for \(i = \text{rightrange, toprange}:\) drawarrow \(\text{origin} -- i\); endfor;
for \(i = 1\) upto \(2\): draw \(g[i]\); endfor;
draw \(f\);
pickup pencircle scaled 1.2pt;
for \(i = 1\) upto \(3\): drawarrow \(x^\ast - - B[i]\); endfor;
endfig; \hfill % 4
\end{verbatim}
input boxes; \% \TeX\ labels are best handled as bordered objects

\% Define shortcuts for drawing arrows from and to boxes.
\vardef cuta(suffix a, b) expr p =
  \drawarrow p cutbefore bpath a cutafter bpath b;
  point .5*length p of p enddef;
\vardef cutb(suffix a, b) expr p =
  \drawarrow p cutbefore bpath a cutafter bpath b;
  point .8*length p of p enddef;
\vardef cutc(suffix a, b) expr p =
  \drawarrow p cutbefore bpath a cutafter bpath b;
  point .2*length p of p enddef;

\% Define self-referential relations.
\vardef loop(suffix a) expr p =
  cuta(a, a)a_{c\{\text{curl}\ #0\} a_{c \{\text{curl}\ #0\}} + p . \{\text{curl}\ #0\} a_{c}} enddef;

beginfig(5);

\% Define some circular objects with \TeX\ labels.
\circleit Knoten[0](btex \ldots etex); \% \TeX\ label: "$0$"
\circleit Knoten[1](btex \ldots etex); \% \TeX\ label: "$1$"
\circleit Knoten[2](btex \ldots etex); \% \TeX\ label: "$2$"
\circleit Knoten[3](btex \ldots etex); \% \TeX\ label: "$3$"
\circleit Knoten[4](btex \ldots etex); \% \TeX\ label: "$\cdots$"

\% Declare the relative position for the labels and draw them.
\% Also, apply self-reference.
for i = 0 upto 3:
  Knoten[i]c = origin + i \times (2.5\, cm, 0); \drawboxed(Knoten[i]);
  label(btex \ldots etex, loop(Knoten[i])(0, 20pt)); \% \TeX\ label: "$\relax$"
endfor;
Knoten[4]c = origin + 4 \times (2.5\, cm, 0);

\% Draw the lower arrows pointing right.
\% Note the use of tension with varying parameters.
for i = 0 upto 3:
  for j = i + 1 upto 4:
    label(btex \ldots etex, cuta(Knoten[i], Knoten[j]) \% \TeX\ label: "$\relax$"
      Knoten[i]c{\{\text{dir}\{(i - j) \times 25\}}} \ldots tension((j - i) \times 1.2 \ldots Knoten[j]c);
  endfor;
endfor;
\% Draw the upper arrows pointing left.
for i = 1 upto 4:
  label(btex \ldots etex, cuta(Knoten[i], Knoten[i - 1]) \% \TeX\ label: "$\relax$"
       Knoten[i]c{\{\text{dir\#65\}}} \ldots Knoten[i - 1]c);
endfor;
endfig; \% Here’s the result:
The material in this MetaPost source file is quite elementary. First a set of nodes (Knoten) are defined and placed, then the arrows are drawn and labelled.

\begin{fig}(6);
\begin{mpl}

\circleit Knoten[0](btex ... etex); Knoten[0]c = origin; \% \TeX\ label: "g"
\circleit Knoten[1](btex ... etex); Knoten[1]c = origin + (2.5cm, 0); \% \TeX\ label: "b"

drawboxed(Knoten[0], Knoten[1]);
\label lft(btex ... etex, loop(Knoten[0])(-20pt, 0)); \% \TeX\ label: "$1-\alpha$"
\label rt(btex ... etex, loop(Knoten[1])(20pt, 0)); \% \TeX\ label: "$1-\beta$"
\label top(btex ... etex, cuta(Knoten[0], Knoten[1])\% \TeX\ label: "$\alpha$"
\label bot(btex ... etex, cuta(Knoten[1], Knoten[0])\% \TeX\ label: "$\beta$"

\end{mpl}
\end{fig}
beginfig(7);

circleit Knoten[0](btex . . . etex); % TeX label: "$0$"
circleit Knoten[1](btex . . . etex); % TeX label: "$1$"
circleit Knoten[2](btex . . . etex); % TeX label: "$2$"
circleit Knoten[3](btex . . . etex); % TeX label: "$\cdots$"
circleit Knoten[4](btex . . . etex); % TeX label: "$r$"

for $i = 0$ upto $4$: Knoten[$i$] = origin + $i \times (1.5cm, 0)$; endfor;
drawboxed(Knoten[0], Knoten[1], Knoten[2], Knoten[4]);
drawunboxed(Knoten[3]);

label lft(btex . . . etex, loop(Knoten[0])(−20pt, 0)); % TeX label: "$r_0$"
label rt(btex . . . etex, loop(Knoten[4])(20pt, 0)); % TeX label: "$r_r$"
label top(btex . . . etex, loop(Knoten[1])(0, 20pt)); % TeX label: "$r_1$"
label top(btex . . . etex, loop(Knoten[2])(0, 20pt)); % TeX label: "$r_2$"
label top(btex . . . etex, cuta(Knoten[0], Knoten[1])); % TeX label: "$p_0$"
label top(btex . . . etex, cuta(Knoten[1], Knoten[2])); % TeX label: "$p_1$"
label top(btex . . . etex, cuta(Knoten[2], Knoten[3])); % TeX label: "$p_2$"
label top(btex . . . etex, cuta(Knoten[3], Knoten[4])); % TeX label: "$p_{r-1}$"
label bot(btex . . . etex, cuta(Knoten[0], Knoten[1])); % TeX label: "$q_0$"
label bot(btex . . . etex, cuta(Knoten[1], Knoten[2])); % TeX label: "$q_1$"
label bot(btex . . . etex, cuta(Knoten[2], Knoten[3])); % TeX label: "$q_2$"
label bot(btex . . . etex, cuta(Knoten[3], Knoten[4])); % TeX label: "$q_r$"

dendfig;

\begin{figure}
\centering
\begin{tikzpicture}
\node[circle, fill, inner sep=1pt] (r) at (0,0) {$r_0$};
\node[circle, fill, inner sep=1pt] (p0) at (1,0) {$p_0$};
\node[circle, fill, inner sep=1pt] (p1) at (2,0) {$p_1$};
\node[circle, fill, inner sep=1pt] (r1) at (3,0) {$r_1$};
\node[circle, fill, inner sep=1pt] (p2) at (4,0) {$p_2$};
\node[circle, fill, inner sep=1pt] (q1) at (5,0) {$q_1$};
\node[circle, fill, inner sep=1pt] (q2) at (6,0) {$q_2$};
\node[circle, fill, inner sep=1pt] (q3) at (7,0) {$q_3$};
\node[circle, fill, inner sep=1pt] (qr) at (8,0) {$q_r$};
\node[circle, fill, inner sep=1pt] (r2) at (9,0) {$r_r$};
\draw[->] (r) edge (p0);
\draw[->] (p0) edge (p1);
\draw[->] (p1) edge (r1);
\draw[->] (r1) edge (p2);
\draw[->] (p2) edge (q1);
\draw[->] (q1) edge (q2);
\draw[->] (q2) edge (q3);
\draw[->] (q3) edge (qr);
\draw[->] (qr) edge (r2);
\end{tikzpicture}
\end{figure}
beginfig(8);

circleit Knoten[0](btex ... etex); % \TeX label: "$0$"
circleit Knoten[1](btex ... etex); % \TeX label: "$1$"
circleit Knoten[2](btex ... etex); % \TeX label: "$2$"
circleit Knoten[3](btex ... etex); % \TeX label: "$ \cdots$"
circleit Knoten[4](btex ... etex); Knoten[4] dx = Knoten[4] dy; % \TeX label: "$r-1$"
circleit Knoten[5](btex ... etex); % \TeX label: "$r$"

for i = 0 upto 5: Knoten[i] c = origin + i * (1.5cm, 0); endfor;

drawboxed(Knoten[0], Knoten[1], Knoten[2], Knoten[4], Knoten[5]);
drawunboxed(Knoten[3]);

label lft(btex ... etex, loop(Knoten[0])(−20pt, 0)); % \TeX label: "$1$"
label rt(btex ... etex, loop(Knoten[5])(20pt, 0)); % \TeX label: "$1$"

for i = 1 upto 4:

    label top(btex ... etex, cuta(Knoten[i], Knoten[i + 1]) % \TeX label: "$p$"
        Knoten[i] c(dir 30) .. Knoten[i + 1] c);

    label bot(btex ... etex, cuta(Knoten[i], Knoten[i − 1]) % \TeX label: "$q$"
        Knoten[i] c(dir 210) .. Knoten[i − 1] c);

endfor;

endfig;

\begin{figure}
\centering
\begin{tikzpicture}
\tikzset{>=latex}
\node (0) at (0,0) {$1$};
\node (1) at (1,0) {$1$};
\node (2) at (2,0) {$2$};
\node (3) at (3,0) {$\cdots$};
\node (4) at (4,0) {$r-1$};
\node (5) at (5,0) {$1$};
\draw (0) edge[bend left] (1)
      (1) edge[bend left] (2)
      (2) edge[bend left] (3)
      (3) edge[bend left] (4)
      (4) edge[bend left] (5);
\end{tikzpicture}
\caption{Diagram of a knot with labeled nodes.}
\end{figure}
beginfig(9);

circleit Knoten[0](btex ... etex); %\TeX\ label: "$0$"
circleit Knoten[1](btex ... etex); %\TeX\ label: "$1$"
circleit Knoten[2](btex ... etex); %\TeX\ label: "$2$"
circleit Knoten[3](btex ... etex); %\TeX\ label: "$\cdots"$
circleit Knoten[4](btex ... etex); %\TeX\ label: "$r$"

for i = 0 upto 4: Knoten[i]c = origin + i * (2cm, 0); endfor;

drawboxed(Knoten[0], Knoten[1], Knoten[2], Knoten[4]);
drawunboxed(Knoten[3]);

label lft(btex ... etex, loop(Knoten[0])(−20pt, 0)); %\TeX\ label: "$\dot{q}_0$"
label top(btex ... etex, cutb(Knoten[0], Knoten[1])) %\TeX\ label: "$\dot{q}_1$"
Knoten[0]c{dir 25} .. {dir −30} Knoten[1]c;
label top(btex ... etex, cutb(Knoten[0], Knoten[2])) %\TeX\ label: "$\dot{q}_2$"
Knoten[0]c{dir 50} .. tension 1.2 .. {dir −30} Knoten[2]c;
label top(btex ... etex, cutb(Knoten[0], Knoten[4])) %\TeX\ label: "$\dot{q}_r$"
Knoten[0]c{dir 75} .. tension 1.44 .. {dir −30} Knoten[4]c;

label bot(btex ... etex, cutc(Knoten[4], Knoten[0])) %\TeX\ label: "$1$"
Knoten[4]c{dir 210} .. tension 1.44 .. {dir 105} Knoten[0]c;
label bot(btex ... etex, cutc(Knoten[2], Knoten[0])) %\TeX\ label: "$1$"
Knoten[2]c{dir 210} .. tension 1.2 .. {dir 130} Knoten[0]c;
label bot(btex ... etex, cutc(Knoten[1], Knoten[0])) %\TeX\ label: "$1$"
Knoten[1]c{dir 210} .. {dir 155} Knoten[0]c;

endfig;

\begin{tikzpicture}
\node (q0) at (0,0) {$q_0$};
\node (q1) at (1,1) {$q_1$};
\node (q2) at (2,1) {$q_2$};
\node (qr) at (3,2) {$q_r$};
\node (q3) at (4,1) {$\cdots$};
\node (q4) at (5,0) {$q_0$};
\node (q5) at (6,0) {$q_1$};
\node (q6) at (7,1) {$q_2$};
\node (q7) at (8,2) {$q_r$};
\node (q8) at (9,1) {$\cdots$};
\node (q9) at (10,0) {$q_0$};

\draw[->] (q0) to (q1);
\draw[->] (q1) to (q2);
\draw[->] (q2) to (qr);
\draw[->] (qr) to (q3);
\draw[->] (q3) to (q4);
\draw[->] (q4) to (q5);
\draw[->] (q5) to (q6);
\draw[->] (q6) to (q7);
\draw[->] (q7) to (q8);
\draw[->] (q8) to (q9);
\end{tikzpicture}
beginfig(10);

boxit Knoten[0][btex ... etex];  % TeX label: "$(0,0)$"
boxit Knoten[1][btex ... etex];  % TeX label: "$(1,0)$"
boxit Knoten[2][btex ... etex];  % TeX label: "$(0,1)$"
boxit Knoten[3][btex ... etex];  % TeX label: "$(1,1)$"

Knoten[0]c = origin;
Knoten[1]c - Knoten[0]c =
    Knoten[3]c - Knoten[2]c = (3cm, 0);
Knoten[0]c - Knoten[2]c = (0, 2cm);

for i = 0 upto 3: drawunboxed(Knoten[i]); endfor;

label top(btex ... etex, cuta(Knoten[0], Knoten[1]))  % TeX label: "$\lambda$"
Knoten[0]c{dir 30} .. Knoten[1]c;

label bot(btex ... etex, cuta(Knoten[1], Knoten[0]))  % TeX label: "$\mu_1$"
Knoten[1]c{dir 210} .. Knoten[0]c;

label lft(btex ... etex, cuta(Knoten[2], Knoten[0]))  % TeX label: "$\mu_2$"
Knoten[2]c -- Knoten[0]c;

label lft(btex ... etex, cuta(Knoten[3], Knoten[1]))  % TeX label: "$\mu_2$"
Knoten[3]c{dir 120} .. Knoten[1]c;

label rt(btex ... etex, cuta(Knoten[1], Knoten[3]))  % TeX label: "$\lambda$"

label bot(btex ... etex, cuta(Knoten[3], Knoten[2]))  % TeX label: "$\mu_1$"

label top(btex ... etex, cuta(Knoten[2], Knoten[3]))  % TeX label: "$\lambda$"

endfig;

\begin{center}
\includegraphics{image.png}
\end{center}
beginfig(11);

boxit Knoten0(btex ... etex); % TeX label: "$(0,0)$"
boxit Knoten1(btex ... etex); % TeX label: "$(0,1)$"
boxit Knoten2(btex ... etex); % TeX label: "$(b,1)$"
boxit Knoten3(btex ... etex); % TeX label: "$(1,0)$"
boxit Knoten4(btex ... etex); % TeX label: "$(1,1)$"

Knoten[1]c = origin;
Knoten[1]c - Knoten[3]c = (0, 2cm);

for i = 0 upto 4: drawunboxed(Knoten[i]); endfor;

label ulft(btex ... etex, cuta(Knoten[1], Knoten[0])) % TeX label: "$\mu_2$"
Knoten[1]c -- Knoten[0]c;

label top(btex ... etex, cuta(Knoten[2], Knoten[1])) % TeX label: "$\mu_2$"

label rt(btex ... etex, cuta(Knoten[3], Knoten[1])) % TeX label: "$\mu_1$"

label wrt(btex ... etex, cuta(Knoten[1], Knoten[4])) % TeX label: "$\lambda$"

label rt(btex ... etex, cuta(Knoten[4], Knoten[2])) % TeX label: "$\mu_1$"

label lft(btex ... etex, cuta(Knoten[0], Knoten[3])) % TeX label: "$\lambda$"
Knoten[0]c -- Knoten[3]c;

label bot(btex ... etex, cuta(Knoten[4], Knoten[3])) % TeX label: "$\mu_2$"

def

endfig;

(0,0) \arrow[<->] (0,1) \arrow[<->] (b,1)
(0,0) \arrow[<->] (1,0) \arrow[<->] (1,1)
beginfig(12);

numeric outer_length; outer_length = 2cm;

pair point[];

point[1] = (0.0, 1.6); point[2] = (0.6, 0.5);
point[3] = (1.7, 0.8); point[4] = (1.7, 2.2);
point[5] = (0.5, 2.8); point[6] = (0.5, 1.9);
point[7] = (1.5, 0.0); point[8] = (2.9, 0.5);
point[9] = (3.0, 2.8); point[10] = (1.5, 3.7);

for i = 1 upto 10:
    point[i] := point[i] * outer_length;
endfor;

point[12] = point[7] + (1.3, 0) * outer_length;


circleit Ecke[1](btex ... etex); % TEx label: "$1$"
circleit Ecke[2](btex ... etex); % TEx label: "$2$"
circleit Ecke[3](btex ... etex); % TEx label: "$3$"
circleit Ecke[4](btex ... etex); % TEx label: "$4$"
circleit Ecke[5](btex ... etex); % TEx label: "$5$"
circleit Ecke[6](btex ... etex); % TEx label: "$6$"
circleit Ecke[7](btex ... etex); % TEx label: "$7$"
circleit Ecke[8](btex ... etex); % TEx label: "$8$"
circleit Ecke[9](btex ... etex); % TEx label: "$9$"
circleit Ecke[10](btex ... etex); % TEx label: "$10$"
circleit Ecke[11](btex ... etex); % TEx label: "$11$"
circleit Ecke[12](btex ... etex); % TEx label: "$12$"
circleit Ecke[13](btex ... etex); % TEx label: "$13$"
circleit Ecke[14](btex ... etex); % TEx label: "$14$"
circleit Ecke[15](btex ... etex); % TEx label: "$15$"
circleit Ecke[16](btex ... etex); % TEx label: "$16$"
circleit Ecke[17](btex ... etex); % TEx label: "$17$"
circleit Ecke[18](btex ... etex); % TEx label: "$18$"
circleit Ecke[19](btex ... etex); % TEx label: "$19$"
circleit Ecke[20](btex ... etex); % TEx label: "$20$"

for i = 1 upto 20:
    Ecke[i].dx = Ecke[i].dy; Ecke[i].c = point[i];
endfor;

pickup pencircle scaled 1pt;

draw Ecke[14].c -- Ecke[19].c; draw Ecke[13].c -- Ecke[9].c;
draw Ecke[18].c -- Ecke[17].c; draw Ecke[3].c -- Ecke[8].c;
draw Ecke[2].c -- Ecke[7].c -- Ecke[12].c;

15

pickup pencircle scaled 4pt;

draw Ecke[17]c -- Ecke[16]c dashed withdots;

pickup pencircle scaled .4pt;

for i = 20 downto 1:
  unfill bpath Ecke[i]; drawboxed(Ecke[i]);
endfor;
endfig;
end