Block Diagrams and Factor Graphs with LaTeX and TikZ

Hans-Andrea Loeliger and Hugo Aguettaz

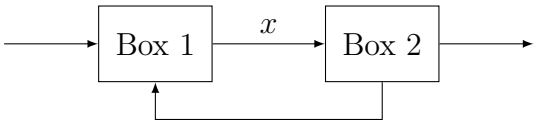
Version 2, December 8, 2025

TikZ is a powerful package for drawing with LaTeX. This document provides some suggestions for a quick start with TikZ for drawing block diagrams and factor graphs.

Contents

1 Block Diagrams

1.1 Getting Started

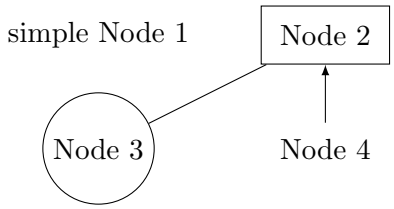



```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\node[rectangle, draw, minimum width=15mm, minimum height=10mm]
     (box1) at (0,0) { Box~1 };
\node[rectangle, draw, minimum width=15mm, minimum height=10mm]
     (box2) at (30,0) { Box~2 };
\draw[->] (box1)+(-20,0) -- (box1);
\draw[->] (box1) -- node[above] {$x$} (box2);
\draw[->] (box2) -- +(20,0);
\draw[->] (box2) -- +(0,-10) -- +(-30,-10) -- (box1);
\end{tikzpicture}
```

Remarks:

- All examples require `\usepackage{tikz}` in the preamble.
- Arguments of `\begin{tikzpicture}`:
  - `x=1mm,y=1mm` changes the unit length from 1cm (default) to 1mm.
  - `>=latex` changes the default arrows to plain-LaTeX arrows.
- The two `\node` commands define nodes with names `box1` and `box2`, respectively. The coordinates refer to the center of the nodes by default.
- The second `\node` command can equivalently be written as
  ```
  \draw (box1)+(30,0)
        node[rectangle, draw, minimum width=15mm, minimum height=10mm]
        (box2) { Box~2 };
  ```
- The last `\draw` command can equivalently be written in many different ways, e.g.:
  - with reversed arrows:
    `\draw[<-] (box1) -- +(0,-10) -- +(30,-10) -- (box2);`
  - with step-by-step relative coordinates:
    `\draw[->] (box2) -- ++(0,-10) -- ++(-30,0) -- (box1);`
  - with automatic corner positioning:
    `\draw[->] (box2) -- +(0,-10) -| (box1);`

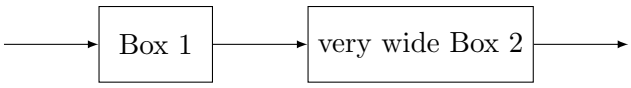
1.2 Positioning and Sizing of Nodes




```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\draw (0,0) node (node1) { simple Node~1 };
\draw (node1)+(30,0) node[draw, inner sep=2.5mm] (node2) { Node~2 };
\draw (node1)+(0,-15) node[circle,draw,inner sep=1mm] (node3) { Node~3 };
\draw (node3-|node2) node[inner sep=2mm] (node4) { Node~4 };
\draw (node3) -- (node2);
\draw[->] (node4) -- (node2);
\end{tikzpicture}
```

Remark: The coordinates of `node4` can equivalently be written as `(node2|-node3)`.

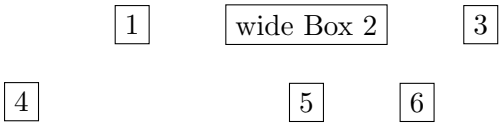
Positioning Relative to Borders:




```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\node[rectangle, draw, minimum width=15mm, minimum height=10mm]
     (box1) { Box~1 };
\node[rectangle, draw, minimum height=10mm, right=12.5mm of box1]
     (box2) { very wide Box~2 };
\draw[->] (box1.west)+(-12.5,0) -- (box1);
\draw[->] (box1) -- (box2);
\draw[->] (box2.east) -- +(12.5,0);
\end{tikzpicture}
```

Remark: The second `\node` command requires `\usetikzlibrary{positioning}`.

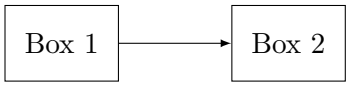
Using Default Border-to-Border Distances:




```
\begin{tikzpicture}[node distance=5mm and 10mm]
\node[draw] (box1) { 1 };
\node[draw, right=of box1] (box2) { wide Box 2 };
\node[draw, right=of box2] (box3) { 3 };
\node[draw, below left=of box1] (box4) { 4 };
\node[draw, below=of box2] (box5) { 5 };
\node[draw, right=of box5] (box6) { 6 };
\end{tikzpicture}
```

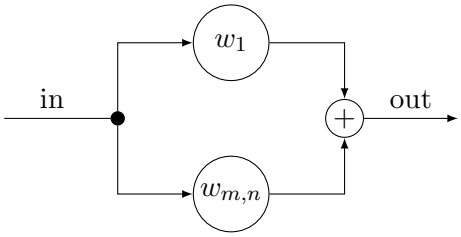
Remark: Requires `\usetikzlibrary{positioning}` in the preamble.

1.3 Predefining Your Standard Nodes




```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\tikzset{
  stdbox/.style = {draw, rectangle, minimum width=15mm, minimum height=10mm},
}
\draw (0,0) node[stdbox] (box1) { Box~1 };
\draw (box1)+(30,0) node[stdbox] (box2) { Box~2 };
\draw[->] (box1) -- (box2);
\end{tikzpicture}
```

More Interesting:



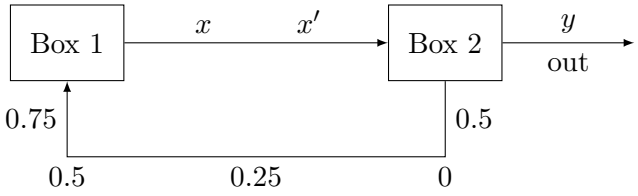
```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\tikzset{
  opcirc/.style = {draw, circle, inner sep = 0, minimum size = 5mm},
  stdcirc/.style = {draw, circle, minimum size = 10mm},
  blobcirc/.style = {draw, fill=black, circle, inner sep = 0, minimum size=1.75mm},
}
\draw (0,0) node[blobcirc] (branching) {};
\draw (branching)+(-15,0) -- (branching) node[pos=0.45, above] {in};
%
\draw (branching)+(15,10) node[stdcirc] (w1) {$w_1$};
\draw[->] (branching) |- (w1);
\draw (branching)+(15,-10) node[stdcirc] (w2) {\makebox(0,0){$w_{m,n}$}};
\draw[->] (branching) |- (w2);
%
\draw (branching)+(30,0) node[opcirc] (adder) {$+$};
\draw[->] (w1) -| (adder);
\draw[->] (w2) -| (adder);
\draw[->] (adder) -- +(15,0) node[pos=0.5, above]{out};
\end{tikzpicture}
```

Remark: The command `\makebox(0,0){$w_{m,n}$}` prevents the circle around $w_{m,n}$ to get too large. That whole line could alternatively be written as
`\draw (branching)+(15,-10) node[stdcirc, label={center:$w_{m,n}$}] (w2) {};`

Nodes with Subpictures:

```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\tikzset{
  pics/crossbox/.style args={#1}{
  code={
    \node[rectangle, draw, inner sep=0mm, minimum width=7.5mm, minimum height=7.5mm]
         (#1) at (0,0) {};
    \draw (#1.south west) -- (#1.north east);
    \draw (#1.north west) -- (#1.south east);
  }
  },
}
\draw (0,0) pic{crossbox={Crossbox1}};
\draw (Crossbox1)+(25,0) pic{crossbox={Crossbox2}};
\draw[->] (Crossbox1) -- (Crossbox2);
\end{tikzpicture}
```

1.4 Positioning Along Paths

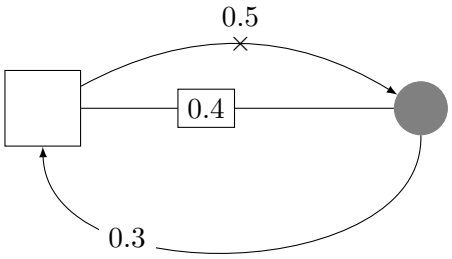



```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\tikzset{
  stdbox/.style = {draw, rectangle, minimum width=15mm, minimum height=10mm},
}
\draw (0,0) node[stdbox] (box1) { Box~1 };
\draw (box1)+(50,0) node[stdbox] (box2) { Box~2 };
\draw[->] (box1) -- node[above,pos=0.3]{$x$} node[above,pos=0.7] {$x'$} (box2);
\draw[->] (box2) -- node[above]{$y$} node[below]{out} +(25,0);
\draw[->] (box2.south) -- node[right,pos=0.5]{0.5} +(0,-10)
         -| (box1) node[below,pos=0]{0} node[below,pos=0.25]{0.25}
                   node[below,pos=0.5]{0.5} node[left,pos=0.75]{0.75};
\end{tikzpicture}
```

Remarks:

- `pos` refers to the fractional position along the path.
- The last `\draw` command comprises two subpaths, each with `pos` ranging from 0 to 1. The second of these subpaths consists of two segments: in the first segment, `pos` ranges from 0 to 0.5; in the second segment, `pos` ranges from 0.5 to 1.

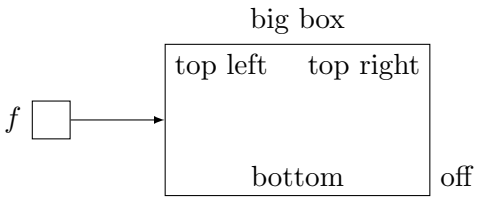
Another Example:



```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\tikzset{
  stdbox/.style = {draw, rectangle, minimum width=10mm, minimum height=10mm},
  stddisc/.style = {draw, circle, fill, minimum size=7mm},
}
\draw (0,0) node[stdbox] (box) {};
\draw (50,0) node[stddisc, color=gray] (disc) {};
\draw[->] (box) to [bend left=30] node {$\times$} node[above=1] { 0.5 } (disc);
\draw (box) -- node[draw, pos=0.4, fill=white] { 0.4 } (disc);
\draw[<-] (box) to [bend right=90] node[pos=0.3, fill=white] { 0.3 } (disc);
\end{tikzpicture}
```

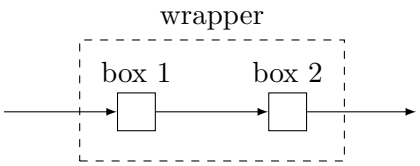
Remark: Note the manual adjustment of the position of "0.5" above "×".

1.5 Labels and Positioning Around Nodes



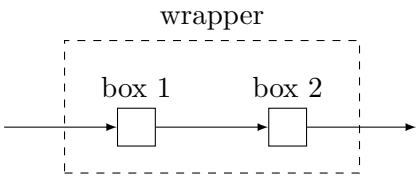

```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\tikzset{
  stdbox/.style = {draw, rectangle, minimum width=5mm, minimum height=5mm},
}
\draw (0,0) node[draw, rectangle, minimum width=35mm, minimum height=20mm,
                 label={above:{big box}}] (Bigbox) {};
\draw (Bigbox.west)+(-15,0) node[stdbox,label={left:$f$}] (f) {};
\draw[->] (f) -- (Bigbox);
\draw (Bigbox.north west) node[anchor=north west] {top left};
\draw (Bigbox.north east) node[anchor=north east] {top right};
\draw (Bigbox.south) node[above] {bottom};
\draw (Bigbox.south east) node[above right] {off};
\end{tikzpicture}
```

Box Around Stuff as a Node, Manual Sizing:




```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\tikzset{
  stdbox/.style = {draw, rectangle, minimum width=5mm, minimum height=5mm},
}
\draw (0,0) node[stdbox,label={above:box~1}] (box1) {};
\draw (box1)+(20,0) node[stdbox,label={above:box~2}] (box2) {};
\draw[->] (box1)+(-17.5,0) -- (box1);
\draw[->] (box1) -- (box2);
\draw[->] (box2) -- +(17,0);
\draw (box1.south west)+(-5,-4) node[rectangle, draw, dashed, anchor=south west,
                  minimum width=35mm, minimum height=16mm, label={above:wrapper}] {};
\end{tikzpicture}
```

Box Around Stuff as a Node, Automatic Sizing:

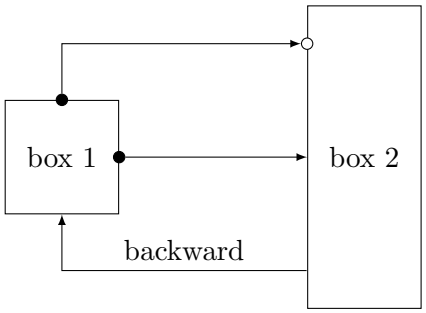



```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\tikzset{
  stdbox/.style = {draw, rectangle, minimum width=5mm, minimum height=5mm},
}
\draw (0,0) node[stdbox,label={[name=_box1] above:box~1}] (box1) {};
\draw (box1)+(20,0) node[stdbox,label={[name=_box2] above:box~2}] (box2) {};
\draw[->] (box1)+(-17.5,0) -- (box1);
\draw[->] (box1) -- (box2);
\draw[->] (box2) -- +(17,0);
\node[draw, dashed, inner sep=3.5mm,
      fit=(box1) (_box1) (box2) (_box2), label={above:wrapper}] {};
\end{tikzpicture}
```

Remarks:

- Requires `\usetikzlibrary{fit}` in the preamble.
- The labels `box1` and `box2` are themselves nodes, with labels `_box1` and `_box2`, respectively.
- Manual sizing may be easier to adjust.

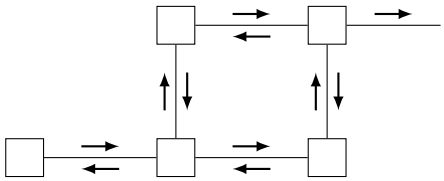
Off-Center Connections:




```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\tikzset{
  empty/.style={circle, minimum size=0mm, inner sep=0mm},
  sbdot/.style={draw, circle, minimum size=1.5mm, fill=black, inner sep=0mm},
  swdot/.style={draw, circle, minimum size=1.5mm, fill=white, inner sep=0mm},
}
\draw (0,0) node[rectangle, draw, minimum width=15mm, minimum height=15mm]
  (box1) { box~1 };
\draw (40,0) node[rectangle, draw, minimum width=15mm, minimum height=40mm]
  (box2) { box~2 };
%
\draw (box1.east) node[sbdot] (box1east) {};
\draw[->] (box1east) -- (box2.west|-box1east);
%
\draw (box1.north) node[sbdot] (box1north) {};
\draw (box2.west)+(0,15) node[swdot] (box2upwest) {};
\draw[->] (box1north) |- (box2upwest);
%
\draw (box2.west)+(0,-15) node[empty] (box2downwest) {};
\draw[->] (box2downwest) -| node[above, pos=0.25]{backward} (box1.south);
\end{tikzpicture}
```

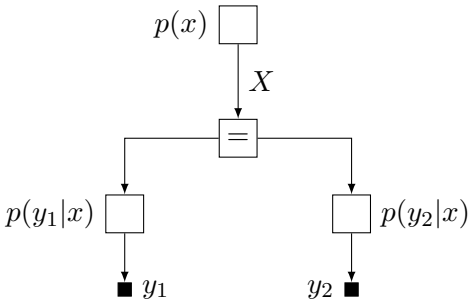
2 Factor Graphs

2.1 Example 1



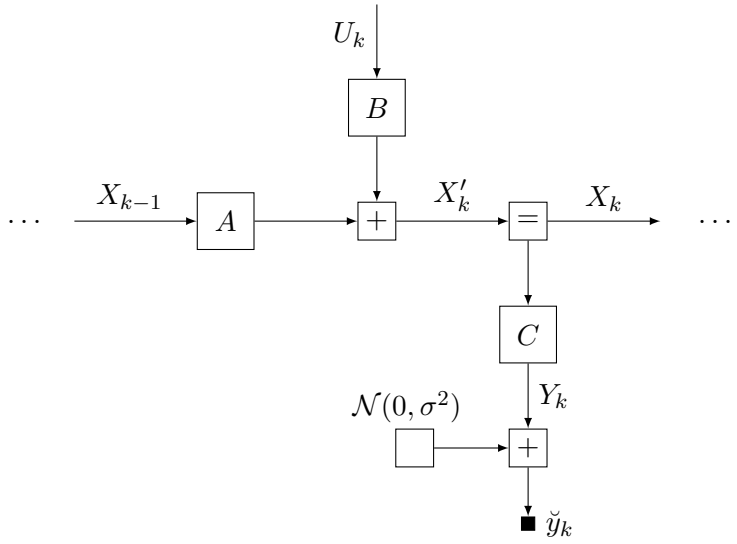
```
\newcommand{\rightmsgarrow}{%
\begin{tikzpicture}
\draw[->, thick] (-2.5,0) -- (2.5,0);
\end{tikzpicture}}

\newcommand{\leftmsgarrow}{%
\begin{tikzpicture}
\draw[<-, thick] (-2.5,0) -- (2.5,0);
\end{tikzpicture}}

\newcommand{\upmsgarrow}{%
\begin{tikzpicture}
\draw[->, thick] (0,-2.5) -- (0,2.5);
\end{tikzpicture}}

\newcommand{\downmsgarrow}{%
\begin{tikzpicture}
\draw[<-, thick] (0,-2.5) -- (0,2.5);
\end{tikzpicture}}

\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\tikzset{ stdbox/.style = {draw, rectangle, inner sep=0mm,
                           minimum width=5mm, minimum height=5mm} }
\draw (0,0) node[stdbox] (box1) {};
\draw (20,0) node[stdbox] (box2) {};
\draw (40,0) node[stdbox] (box3) {};
\draw (20,17.5) node[stdbox] (box4) {};
\draw (40,17.5) node[stdbox] (box5) {};
\draw (box4) -- node[above] {\rightmsgarrow} node[below] {\leftmsgarrow} (box5);
\draw (box5) -- node[above] {\rightmsgarrow} +(15,0);
\draw (box4) -- node[left] {\upmsgarrow} node[right] {\downmsgarrow} (box2);
\draw (box5) -- node[left] {\upmsgarrow} node[right] {\downmsgarrow} (box3);
\draw (box1) -- node[above] {\rightmsgarrow} node[below] {\leftmsgarrow} (box2);
\draw (box2) -- node[above] {\rightmsgarrow} node[below] {\leftmsgarrow} (box3);
\end{tikzpicture}
```

2.2 Example 2




```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\tikzset{
  opbox/.style = {draw, rectangle, inner sep=0mm,
                  minimum width=5mm, minimum height=5mm},
  blobbox/.style = {draw, fill=black, rectangle, inner sep=0mm,
                    minimum width=1.75mm, minimum height=1.75mm},
}
\draw (0,0) node[opbox] (equ) {$=$};
\draw (equ)+(0,15) node[opbox, label={left: $p(x)$}] (px) {};
\draw[->] (px) -- node[right]{$X$} (equ);
%
\draw (equ)+(-15,-10) node[opbox, label={left: $p(y_1 | x)$}] (py1cx) {};
\draw[->] (equ) -| (py1cx);
\draw (py1cx)+(0,-10) node[blobbox, label={right: $y_1$}] (y1) {};
\draw[->] (py1cx) -- (y1);
%
\draw (equ)+(15,-10) node[opbox, label={right: $p(y_2 | x)$}] (py2cx) {};
\draw[->] (equ) -| (py2cx);
\draw (py2cx)+(0,-10) node[blobbox, label={left: $y_2$}] (y2) {};
\draw[->] (py2cx) -- (y2);
\end{tikzpicture}
```

2.3 Example 3




```
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\tikzset{
  opbox/.style = {draw, rectangle, inner sep=0mm,
                  minimum width=5mm, minimum height=5mm},
  stdbox/.style = {draw, rectangle, minimum width=7.5mm, minimum height=7.5mm},
  blobbox/.style = {draw, fill=black, rectangle, inner sep=0mm,
                    minimum width=1.75mm, minimum height=1.75mm},
}
\draw (0,0) node[stdbox] (A) {$A$};
\draw[->] (A)+(-20,0) -- (A) node[pos=0.45, above]{$X_{k-1}$};
\draw (A.west)+(-22.5,0) node { \ldots };
\draw (A)+(20,0) node[opbox] (plus) {$+$};
\draw[->] (A) -- (plus);
%
\draw (plus)+(0,15) node[stdbox] (B) {$B$};
\draw[->] (B) -- (plus);
\draw (B)+(0,15) node (inpU) {};
\draw[->] (inpU) -- (B) node[pos=0.4, left] {$U_k$};
%
\draw (plus)+(20,0) node[opbox] (equ) {$=$};
\draw[->] (plus) -- (equ) node[pos=0.5, above] {$X'_k$};
\draw (equ)+(0,-15) node[stdbox] (C) {$C$};
\draw[->] (equ) -- (C);
\draw (C)+(0,-15) node[opbox] (outplus) {$+$};
 \draw (outplus)+(-15,0) node[opbox] (obsnoise) {};
 \draw[->] (obsnoise) -- (outplus);
```

```
 \draw (obsnoise)+(-1,2) node[above] {$\mathcal N(0, \sigma^2)$};
\draw[->] (C) -- (outplus) node[pos=0.5, right] {$Y_k$};
\draw (outplus)+(0,-10) node[blobbox] (blob) {};
\draw[->] (outplus) -- (blob);
\draw (blob)+(1,0) node[right] {$\breve y_k$};
%
\draw[->] (equ)-- +(17.5,0) node[pos=0.5, above] {$X_k$};
\draw (equ.east)+(22.5,0) node { \ldots };
\end{tikzpicture}
```

2.4 LaTeX Commands for Messages and Message Parameters

The commands `\msgf` and `\msgb` provided below allow to write $\vec{\mu}_X, \overleftarrow{\mu}_X, \vec{V}_Z, \overleftarrow{m}_Y$ as `$\msgf{\mu}{X}, \msgb{\mu}{X}, \msgf{V}{Z}, \msgb{m}{Y}$`.

```
% Definition of \msgf and \msgb for messages and message parameters
% by Christoph Reller
\makeatletter
\DeclareFontFamily{U}{MnSymbolA}{}
\DeclareSymbolFont{MnSyA}{U}{MnSymbolA}{m}{n}
\DeclareFontShape{U}{MnSymbolA}{m}{n}{
<-6> MnSymbolA5
<6-7> MnSymbolA6
<7-8> MnSymbolA7
<8-9> MnSymbolA8
<9-10> MnSymbolA9
<10-12> MnSymbolA10
<12-> MnSymbolA12}{}
\DeclareMathSymbol{\smallrightarrow}{\mathrel}{MnSyA}{0}
\DeclareMathSymbol{\smallleftarrow}{\mathrel}{MnSyA}{2}
\DeclareMathSymbol{\smallleftrightarrow}{\mathrel}{MnSyA}{16}
\newcommand{\smallrightarrowfill@}{\arrowfill@\relbar\relbar\smallrightarrow}
\newcommand{\smallleftarrowfill@}{\arrowfill@\smallleftarrow\relbar\relbar}
\newcommand{\smallleftrightarrowfill@}
{\arrowfill@\smallleftarrow\relbar\smallrightarrow}
\renewcommand{\overrightarrow}{\mathpalette{\overarrow@\smallrightarrowfill@}}
\renewcommand{\overleftarrow}{\mathpalette{\overarrow@\smallleftarrowfill@}}
\renewcommand{\overleftrightarrow}
{\mathpalette{\overarrow@\smallleftrightarrowfill@}}
\makeatother
\providecommand{\msgf}[2]{\protect\overrightarrow{#1}_{\mspace{-3mu}#2}}
\providecommand{\msgb}[2]{\protect\overleftarrow{#1}_{\mspace{-3mu}#2}}
```

3 Moving On

3.1 Standalone Graphics

It often makes sense to create graphics as separate files. That applies, in particular, to nontrivial plots, or plots of nontrivial data. Such external files can be included in the main LaTeX source using the `standalone` package, e.g.,

```
\includestandalone[mode=buildnew]{my_precious_graphics}
```

The source file `my_precious_graphics.tex` can use the `standalone` document class, e.g.,

```
\documentclass[tikz]{standalone}
\begin{document}
\begin{tikzpicture}[x=1mm, y=1mm, >=latex]
\draw (0,0) node[rectangle, draw, minimum width=15mm, minimum height=10mm] { };
\end{tikzpicture}
\end{document}
```

3.2 Links to Resources on LaTeX and TikZ

LaTeX:

- https://www.latex-project.org/
- https://tobi.oetiker.ch/lshort/lshort.pdf
- https://moser-isi.ethz.ch/manuals.html#eqlatex
- https://latex-tutorial.com/color-latex/

Easy Introductions to TikZ:

- https://rmwu.github.io/tutorial/latex/2019/08/27/intro/
- https://tikz.org/examples/chapter-02-creating-the-first-tikz-images/
- https://latexdraw.com/draw-a-rectangle-in-tikz/#t-1618533406677
- https://tikz.org/examples/chapter-03-drawing-positioning-and-aligning-nodes/
- https://rmwu.github.io/tutorial/latex/2019/11/21/positioning/

TikZ and PGF Manual:

- https://pgf-tikz.github.io/pgf/pgfmanual.pdf
- https://tikz.dev/
- https://github.com/pgf-tikz/pgf