Alexander R. Perlis

Abstract

By default, balance-pic aligns diagrams according to the center of each object. For many types of diagrams, such center alignment is the preferred choice; however, axis alignment is sometimes better. For example, compare $A \to A^2$ (center-aligned) with $A \to A^2$ (axis-aligned); the arrow stayed in the same place, but the $A$ moved up a little, and the $A^2$ moved up a lot. Note that the simple TeX code $\$A \to A^2\$ uses axis alignment: $A \to A^2$.

This article studies attempts to instruct balance-pic to use axis alignment and presents a concise solution to the problem. Enhancements to balance-pic are proposed.

1 Preliminaries

The version of balance-pic used here is 3.7 (16Feb1999). The balance-pic User’s Guide will be cited as [XY GUIDE], and the balance-pic Reference Manual as [XY MANUAL]. These documents are part of the balance-pic distribution available on CTAN.

This article’s abstract already defined center alignment and exhibited the subtle differences between that and axis alignment, but the latter was left undefined. For now, it means “the alignment used by the simplest of TeX code”. Matters will make more sense after section 5, where the alignment practices of TeX and balance-pic are explained in detail.

For most of this article, we will study attempts at axis alignment using the \texttt{\xymatrix} feature of balance-pic. (It is introduced nicely in [XY GUIDE].) The one-line solution to our alignment problem appears in section 7. Solutions for the \texttt{\xygraph} feature and for balance-pic kernel code appear there as well.

2 The problem

We are (happily) compelled to use balance-pic because it produces fantastic diagrams we cannot otherwise obtain, yet we recognize that many of our center-aligned diagrams ought to be axis-aligned. We wish to instruct balance-pic to use the preferred alignment.

To study the matter, we need a toy example that exhibits a wide gulf between the two types of alignment, yet fits in this article’s columns.

\begin{verbatim}
def\toyexample{ \%}
def\toyone{c} \def\toytwo{a_{\frac{x}{y}}} 
def\toythree{\hat{\underline{\underline{g}}}} 
def\toyfour{\hat{\hat{\overline{h^2}}}}
\end{verbatim}

The result of \texttt{\xymatrix{\toyexample}} is

\begin{align*}
&\xymatrix{\toyone \ar[r] & \toytwo \\
&\toythree \\
&\toyfour & \}
\end{align*}

What a monstrosity! To shirk responsibility, let’s take the viewpoint that a famous mathematician has hired us to typeset this bewildering diagram as part of her new book. The notation is beyond our control.

A quick peek (go ahead!) at the end of section 7 shows what we’re after. To get a sense of the difference, compare the top rows of the two diagrams without the arrows:

\begin{align*}
ca_{\frac{x}{y}} g & \overset{h^2}{\sim} \\
ca_{\frac{x}{y}} & \overset{h^2}{\sim}
\end{align*}

The reader interested merely in the solution to our problem should skip directly to section 7, or to section 5 for some background on that solution.

The material below in sections 3 and 4 has nothing to do with the ultimate solution, and is included mainly for the balance-pic enthusiast interested in why \texttt{\xymatrix\01} and variations on that theme do not solve our problem.

3 Using \texttt{\xymatrix\01}

The first idea is to try \texttt{\xymatrix\01} in place of \texttt{\xymatrix}. This gives

\begin{align*}
&\xymatrix{c \ar[r] & a_{\frac{x}{y}} \\
&g \\
&\frac{g}{h^2} & \}
\end{align*}

The top row is

\begin{align*}
ca_{\frac{x}{y}} g & \overset{h^2}{\sim}
\end{align*}

That’s neither center-aligned nor axis-aligned! To be fair, [XY GUIDE, §1.4, p.3] only encourages use of \texttt{\xymatrix\01} with one-line diagrams. Leaving our toy example aside for the rest of this section, let’s...
experiment a bit with one-line diagrams:

\[ \xymatrix{A \ar[r] & A'} \]

\[ \xymatrix{A \ar[r] & A^2} \]

Hoohrah! The latter is axis-aligned! But if we replace $A'$ with $A^2$, we get:

\[ \xymatrix{A \ar[r] & A'^2} \]

Unhoorah. This time the result is neither center-aligned nor axis-aligned: the placement of $A^2$ is too low. The mistake is easier to spot by enlarging and putting all the diagrams on a rule:

$A \rightarrow A'$ $A \rightarrow A'$ $A \rightarrow A'$

$A \rightarrow A^2$ $A \rightarrow A^2$ $A \rightarrow A^2$

Left: $\xymatrix$. Middle: $\xymatrix@1$. For comparison, \$A \to A'\$ and \$A \to A'^2\$ are included on the right—being the simplest of \TeX code, they are axis-aligned by definition!

In one case of a one-line diagram, \xymatrix@1 succeeds, yet in another, it fails. Why didn’t that happen? The difference between $\xymatrix$ and \xymatrix@1 is that the latter inserts a zero-width left parenthesis at the start of every entry, and doing so affects the vertical spacing.\footnote{There is another difference between $\xymatrix$ and \xymatrix@1, not documented in [XY MANUAL]: @1 implies \texttt{OM=1pt}. This explains why, in comparing the diagram at the start of section 3 to the one at the end of section 2, the arrows are closer to the entries.}

\[ \xymatrix{A \ar[r] & (A')} \]

\[ \xymatrix{A \ar[r] & (A^2)} \]

An entry’s center is determined by its bounding box, which in turn is determined by the parts of the entry that stick out the most. Both $A$ and $A'$ are dwarfed by the parenthesis, but $A^2$ is not: the superscript sticks out more than the parenthesis. Consequently, the center of $(A^2)$ is slightly higher than that of $(A$ or $(A'$; thus, to align entries by their center, X\TeX-pic must lower $(A^2$ slightly. (Why doesn’t it instead raise everything else? We’ll answer this question in section 5.)

In summary, \xymatrix@1 gives an axis-aligned result only when the diagram’s entries fit inside regular parentheses. Otherwise, the result likely will be neither center-aligned nor axis-aligned.

\section{Mimicking $\xymatrix$}

Ah ha! With large entries, the zero-width left parenthesis inserted by \xymatrix@1 is not tall enough, so let’s use a taller one. Since \xymatrix@1{} is equivalent\footnote{The diagrams in \$\ldots\$ were made smaller by setting up \xymatrix with \texttt{OM=1pt}. The sample code does not reflect this.} to

\[ \\xymatrix{ & } \]

we might first try

\[ \\xymatrix{ & } \]

but discover this isn’t enough, and after running out of named sizes, we might try letting \TeX make the precise calculation:

\[ \\xymatrix{ & } \]

Hey, this is axis-aligned! Unfortunately, many of the arrows now appear to be afraid of the entries. To understand what went wrong, take a look at the size of the delimiter we inserted around each entry:

\[ \left( \xymatrix@1{toyexample} \right. \]

\[ \left( \xymatrix@1{toyexample} \right. \]

Each entry’s vertical size is determined by the delimiter, and then, as usual, \xymatrix adds an additional margin:

Evidently, the promising approach of inserting zero-width material to affect vertical alignment, which is used by \xymatrix@1, is \textit{fundamentally flawed}: important height information gets lost!

It’s time to step back and study the alignment algorithms in \TeX and X\TeX-pic.
5 How \TeX{} and \XYPic align objects

Earlier we asked: in going from $A \rightarrow A'$ to $A' \rightarrow A'$, why does the arrow stay put and the $A$ and $A'$ move up, instead of, say, the $A$ staying put, the arrow moving down, and the $A'$ moving (slightly) up?

Inside math mode, \TeX{} maintains two reference lines for alignment purposes: the baseline and the axis. These lines are not part of the characters being typeset; rather, they depend only on the current font and thus should be thought of as being part of the underlying canvas. As for the characters, each one has a bounding box and a reference point (obtained from the font’s TFM file).

In horizontal mode and in math mode, \TeX{} positions each character so that the character’s reference point lands on the canvas’s baseline. The axis comes into play when \TeX{} builds a fraction: the numerator and denominator are positioned so that the bar of the fraction lands on the canvas’s axis. Each delimiter, such as the left parenthesis, is designed to involve both lines: as with all characters, \TeX{} positions the delimiter so that its reference point lands on the canvas’s baseline; however, in so doing, due to the shape of the delimiter, the middle of the delimiter lands precisely on the canvas’s axis. In other words, after placement, each delimiter has equal height and depth when measured from the axis, but not when measured from the baseline.

\XYPic, on the other hand, maintains its own reference point for each object, which starts out in the center of the object. When \XYPic hands a finished object to \TeX{} for placement, it does so in such a way that the object’s \XYPic reference point lands on the canvas’s axis.\footnote{As explained in \cite[§2.1, p.6]{xymanual}, that’s true only when \XYPic was entered inside math mode. Otherwise there is no axis, and so the canvas’s baseline is used. This affects everything. The net effect is to shift the entire diagram, arrows and all, by a fixed amount.}

Let’s summarize. When \TeX{} is in charge, the \TeX{} reference point lands on the baseline. When \XYPic is in charge, the \XYPic reference point lands on the axis.

Let’s illustrate. On the left, we see how \TeX{} positions an $A$: the \TeX{} reference point • lands on the baseline.

Next we see how \XYPic positions $A$, $(A)$, and $(A^2)$: the \XYPic reference point ◦, which defaults to the object’s center, lands on the axis. In the case of $(A)$, the \TeX{} reference point happens to land on the baseline, but that’s merely a consequence of how delimiters are designed (discussed earlier in this section).

We wish to achieve the following:

\begin{center}
\begin{tikzpicture}
\matrix (X) [matrix of math nodes, row sep=2em, column sep=2em, text height=1.5ex, text depth=0.25ex]
{ A \quad (A) \quad (A^2) }
\end{tikzpicture}
\end{center}

The \XYPic reference point should be positioned away from the center in such a way that when it lands on the axis, the \TeX{} reference point will land on the baseline. The crucial measurement is easy to spot: the vertical distance between the \XYPic reference point and the \TeX{} reference point should be the same as the distance between the canvas’s baseline and axis. \TeX{} will cough up that distance if you feed it \texttt{\\fontdimen22\textfont2}\texttt{\textfont2}.

By positioning the \XYPic reference point appropriately, we achieve the desired alignment without changing the object’s bounding box. This is the solution we’ve been seeking! The code is presented in section 7.

6 Aside: the term axis-aligned

By the discussion in the previous section, we conclude that axis-aligned means: objects are aligned with their \TeX{} reference point on the baseline, while diagram arrows point to the axis (because that’s where the \XYPic reference point is). But the following description does a better job justifying the term. Before being dropped on the canvas, each object is typeset in its own box and thus has its own axis. Getting the object’s \TeX{} reference point onto the canvas’s baseline is equivalent to getting the object’s axis onto the canvas’s axis. Thus axis-aligned means: each object’s axis lies on the canvas’s axis, and each arrow points to that common axis. In short, everything is aligned by the axis!

7 The solution

We have seen that \XYPic positions an object so that the \XYPic reference point, which defaults to the object’s center, lands on the canvas’s axis. To alter the placement, we either move the object’s center by changing its size, or move the \XYPic reference point away from the center. In section 4 we disposed of the idea of changing the object’s size, because the original size is needed later for drawing arrows. At the end of section 5, we saw that moving the \XYPic reference point appropriately solves our problem.

All said and done, our solution is to put \texttt{\entrymodifiers={+!!<0pt,+.snap to grid,\fontdimen22\textfont2}}\texttt{\textfont2} prior to each \texttt{\xymatrix}, or simply once and for all in the document’s preamble.
The \+ sets up a margin similar to the default margin used by \texttt{\xymatrix} (the difference is discussed briefly in section 9). The first \! moves the \texttt{XY-pic} reference point from the object’s center down to the line containing \texttt{\LaTeX}'s reference point, and then \!\texttt{<0pt,\fontdimen22\textfont2>} moves it up the appropriate distance so that, when it is dropped on the canvas’s axis, the \texttt{\LaTeX} reference point lands on the canvas’s baseline.

Although \texttt{[XY GUIDE]} introduces \texttt{XY-pic} in terms of the \texttt{\xymatrix} feature, there are other ways of using \texttt{XY-pic}, notably \texttt{xygraph} or even direct kernel code. With the \texttt{xygraph} feature, axis alignment is obtained by putting
\begin{verbatim}
!~*{+!!<0pt,\fontdimen22\scriptfont2>}
\end{verbatim}
at the start of each graph. With \texttt{XY-pic} kernel code, add the drop modifiers
\begin{verbatim}
!!<0pt,\fontdimen22\textfont2>
\end{verbatim}
to each object that should be axis-aligned. But be careful: the effect is cumulative. Thus, if you drop a \texttt{\composite} of objects that should be axis-aligned, either axis-align the composite object, or the individual objects, but not both.

Returning to our toy example from section 2, the axis-aligned result is:
\begin{verbatim}
\entrymodifiers={+!!<0pt,\fontdimen22\textfont2>}
\xymatrix{\toyexample}
\end{verbatim}

\begin{diagram}
\begin{array}{c}
c \\
\downarrow z \\
H^2 \\
\end{array}
\begin{array}{c}
\rightarrow a \\
\rightarrow g \\
\end{array}
\begin{array}{c}
y \\
\rightarrow g \\
\end{array}
\end{diagram}

\section{Caveats}
\subsection{Size}
By default, \texttt{XY-pic} builds objects in \texttt{\textstyle}.
\entrymodifiers={+!!A} in the preamble, but newcomers to XY-pic are likely to master the use of \texttt{xymatrix} setups prior to tackling kernel-level drop modifiers.

5. Similarly, the graph option might support \texttt{\everyxygraph={...}}. When missing, the usual defaults [XY \textit{manual}, p.53] would apply.

10 Acknowledgments

The problem of aligning objects and arrows appropriately to achieve beautiful diagrams is hardly new. (Just think of \TeX itself, or all the diagram packages available on CTAN.) Within the context of XY-pic, the problem was discussed and solved in 2001 on the XY-pic list by Vadim Radionov, whose solution is essentially identical to the one obtained here.

I thank Ross Moore and Florian Lengyel for commenting on earlier versions of this article, and Michael Abbott for discussing the ideas in section 9.

\begin{itemize}
\item Alexander R. Perlis \\
Department of Mathematics \\
The University of Arizona \\
Tucson, AZ 85721 USA \\
aprl@math.arizona.edu
\end{itemize}