Towards An Operational (La)TEX Package Supporting Optical Scaling of Dynamic Mathematical Symbols

Abdelouahad BAYAR
a.bayar@uca.ma

Cadi Ayyad University
École Supérieure de Technologie de Safi
(High College of Technology)

July 27, 2016

- Handling of dynamic mathematical symbols is still a hard problem
- (Some)TEX and PostScript Type 3 to supply a solution
- A TEX (luaTEX) package supporting optical scaling and supplying the habitual way to format scientific documents

- Static mathematical symbols
- Variable sized symbols (Dynamic mathematical symbols)

- Linear scaling

$$\left\{ a_{11}x_1 + \cdots \right. \quad \left\{ \begin{array}{l} a_{11}x_1 + \cdots \\ a_{21}x_1 + \cdots \end{array} \right. \quad \left\{ \begin{array}{l} a_{11}x_1 + \cdots \\ a_{21}x_1 + \cdots \\ a_{31}x_1 + \cdots \end{array} \right. \quad \left\{ \begin{array}{l} a_{11}x_1 + \cdots \\ a_{21}x_1 + \cdots \\ a_{31}x_1 + \cdots \\ a_{41}x_1 + \cdots \end{array} \right.$$

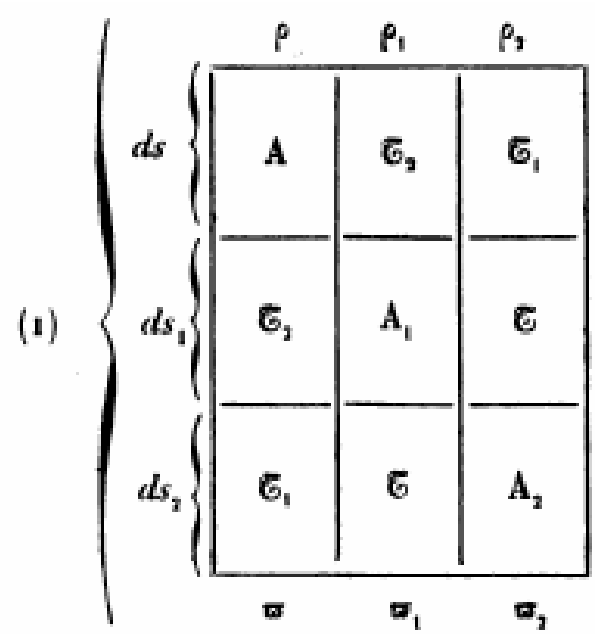
- Linear scaling

$$\left\{ \begin{array}{l} a_{11}x_1 + \cdots \\ a_{21}x_1 + \cdots \end{array} \right. \quad \left\{ \begin{array}{l} a_{11}x_1 + \cdots \\ a_{21}x_1 + \cdots \\ a_{31}x_1 + \cdots \end{array} \right. \quad \left\{ \begin{array}{l} a_{11}x_1 + \cdots \\ a_{21}x_1 + \cdots \\ a_{31}x_1 + \cdots \\ a_{41}x_1 + \cdots \end{array} \right.$$

- Optical scaling

$$\left\{ \begin{array}{l} a_{11}x_1 + \cdots \\ a_{21}x_1 + \cdots \end{array} \right. \quad \left\{ \begin{array}{l} a_{11}x_1 + \cdots \\ a_{21}x_1 + \cdots \\ a_{31}x_1 + \cdots \end{array} \right. \quad \left\{ \begin{array}{l} a_{11}x_1 + \cdots \\ a_{21}x_1 + \cdots \\ a_{31}x_1 + \cdots \\ a_{41}x_1 + \cdots \end{array} \right.$$

- Metal typesetting

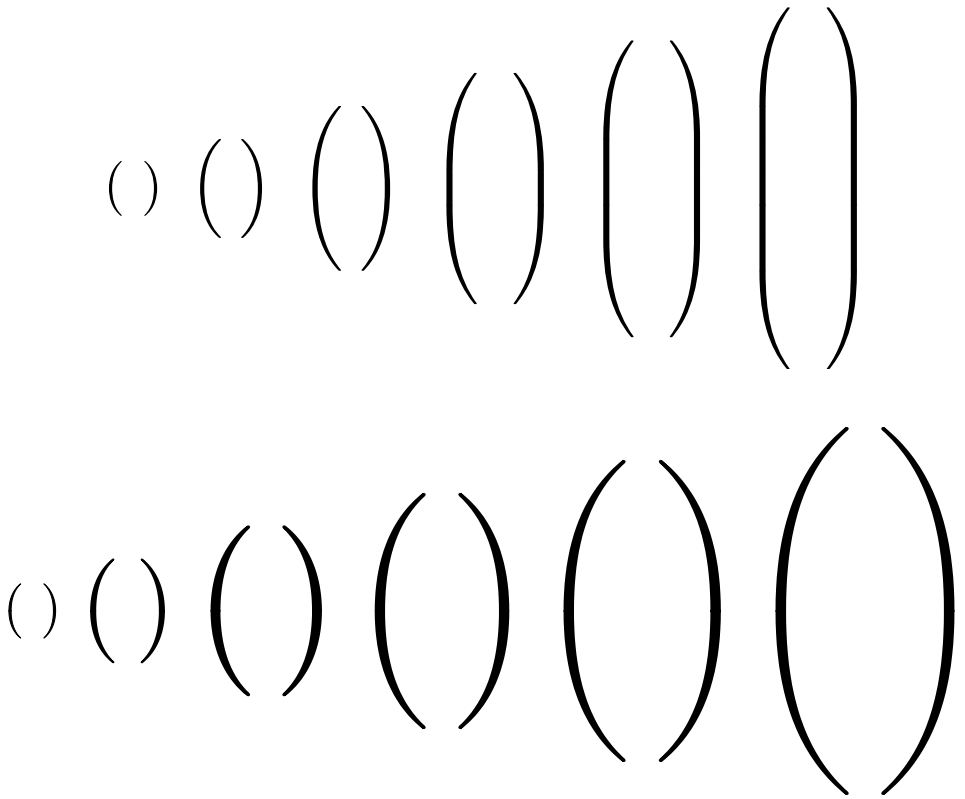


- Digital typesetting

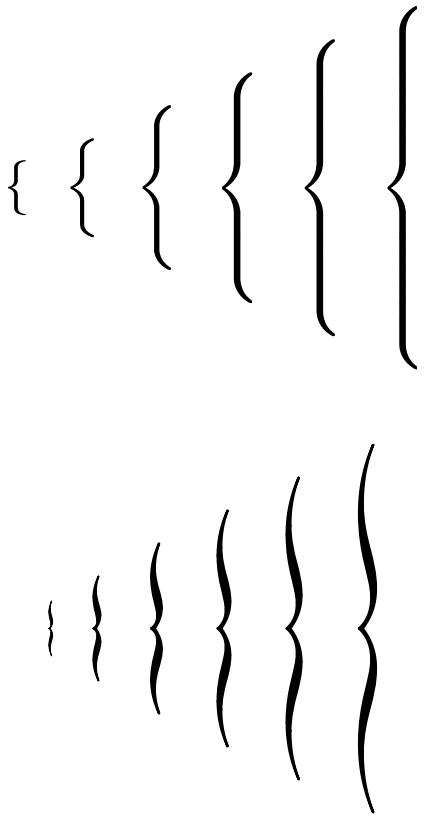
- math-fly/Grif
- Curext
- TEX/LATEX
  - Support of mathematical formulas with multiple variable-sized symbols

$$\left(\begin{array}{cc} \left(\begin{array}{cc} a & b \\ c & d \end{array} \right) & \left(\begin{array}{cc} e & f \\ g & h \end{array} \right) \\ 0 & \left(\begin{array}{cc} i & j \\ k & l \end{array} \right) \end{array} \right)$$

- TEX/LATEX
  - The thickness remains constant after some level
  - Some symbols differ from corresponding ones in metal typesetting with regard to the shape after a level of size



- TEX/LATEX
  - Some symbols differ from their corresponding in metal typesetting with regard to the shape at all sizes.



- Font requirements
- Document processing system requirements
- use Requirements by end-users
- Required Font for (La)\TEX: PostScript Type 3

dynMath.tex

```
⋮
\special{!
⋮ some useful PostScript procedures
/w 0 def
/h 0 def
/fs 0 def
/dynMathFont 8 dict def
dynMathFont begin
/FontType 3 def
/FontMatrix [0.001 0 0 0.001 0 0] def
/FontBBox [ 0 0 1000 1000] def
/Encoding 256 array def
0 1 255 {Encoding exch /.notdef put} for
```

dynMath.tex

```
Encoding 0 /leftPar put % Left parenthesis = 0
Encoding 1 /rightPar put % Right parenthesis = 1
Encoding 8 /leftBrace put % Left brace = 8
Encoding 9 /rightBrace put % Right brace = 9
⋮
⋮/Charprocs - individual glyph descriptions
⋮
⋮/BuildGlyph definition
⋮
⋮/BuildChar defintion
⋮
```

dynMath.tex

```
currentdict
end % End of font dictionary
/dynMath exch definefont pop%
}
⋮
⋮ Some useful TEX defintions
⋮
% #1:left delimiter, #2:formula, #3:right delimiter
\def\meLeft#1#2\meRight#3{
⋮ macro defintion
}
```

\meLeft macro

The macro \meLeft has to :

1. Compute the dimensions of the formula (in the correct math style): \hf, \df and \wf
2. Determine vertical and horizontal stretching amounts: \h and \w
3. Calculate the size of the font \fs in which dynMath will be used
4. Determine the dimensions of the left symbol: \symHeight, \symWidth and \symDepth

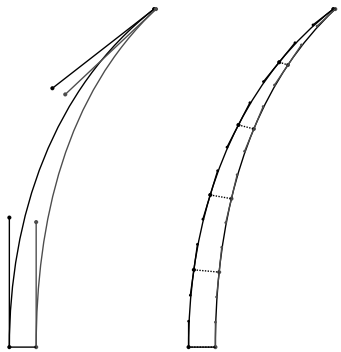
\meLeft macro

5. Build an horizontal box \efbDel:
```
\setbox\efbDel\hbox {
  \special{" ...
    /fs ...  store
    /h ...  store
    /w ...  store
    /dynMath findfont fs scalefont setfont
    <code of The left symbol> show
  }
}
```
6. Set the dimensions of \efbDel to \symHeight, \symWidth and \symDepth
7. Write the content of \efbDel

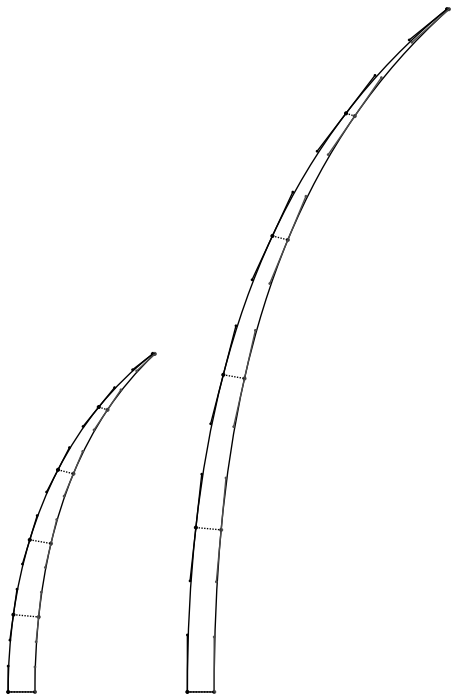
\meLeft macro

8. Write the mathematical formula
9. Apply steps from the second to the seventh to deal with the right delimiter

Symbol parametrizing stretching - Concepts

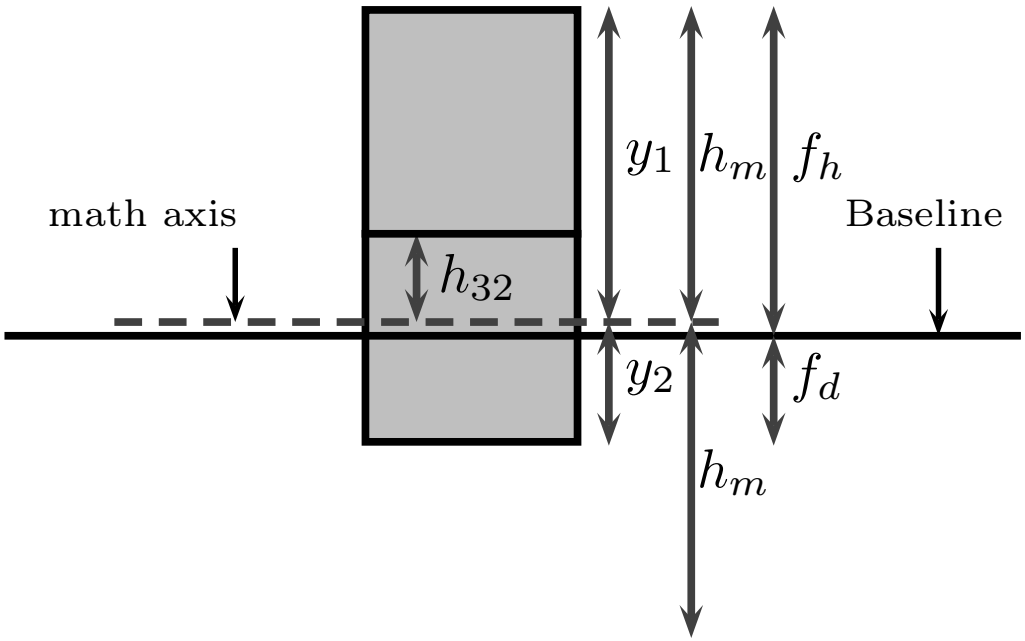


Symbol parametrizing stretching - Concepts



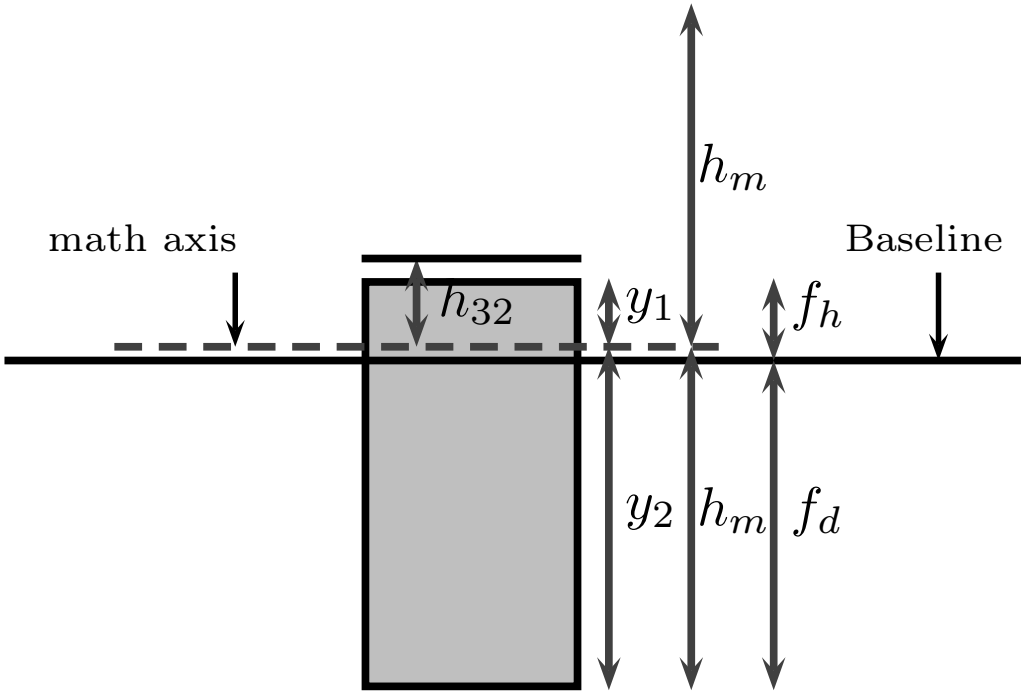
Mathematical formula characteristics

- High mathematical formula



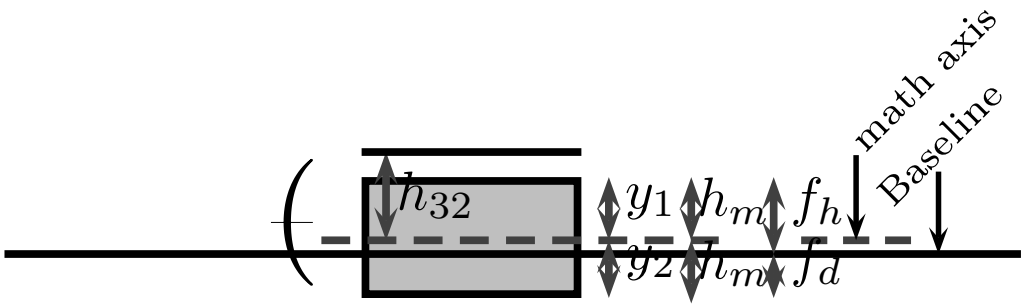
Mathematical formula characteristics

- Deep mathematical formula



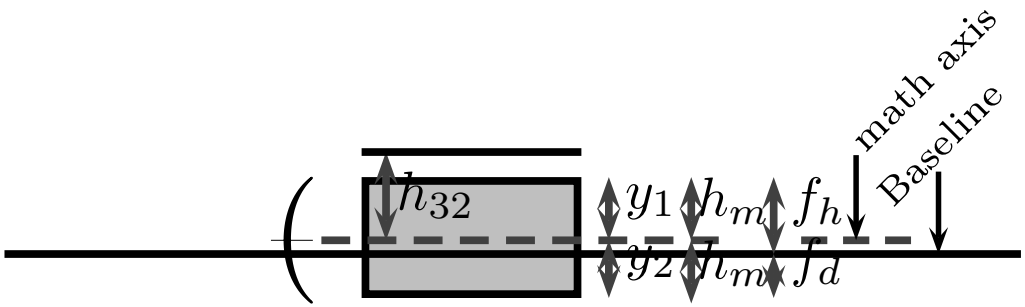
Scaling - $h_m \leq h_{32}$

$$fs = \frac{1003.75 \times h_m}{h_{1000}}$$

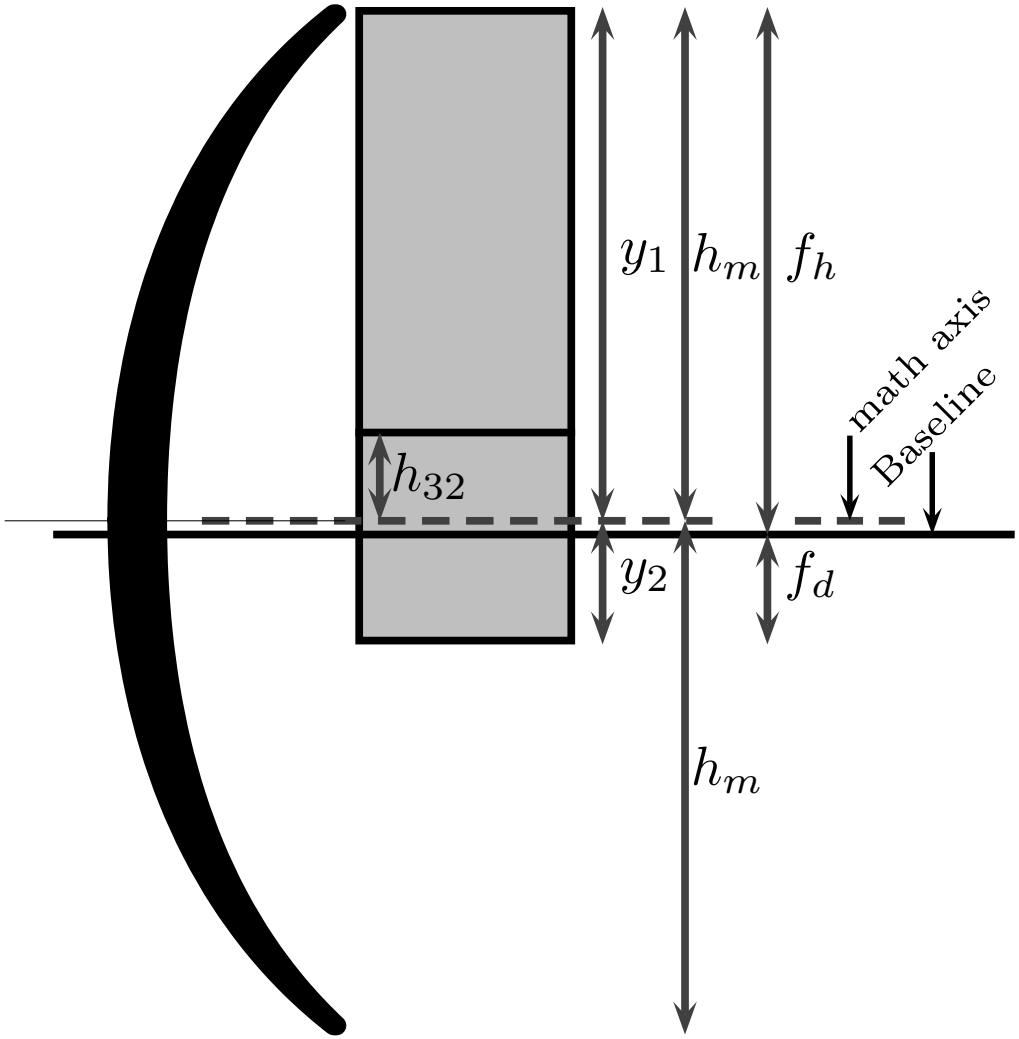


Scaling - $h_m \leq h_{32}$

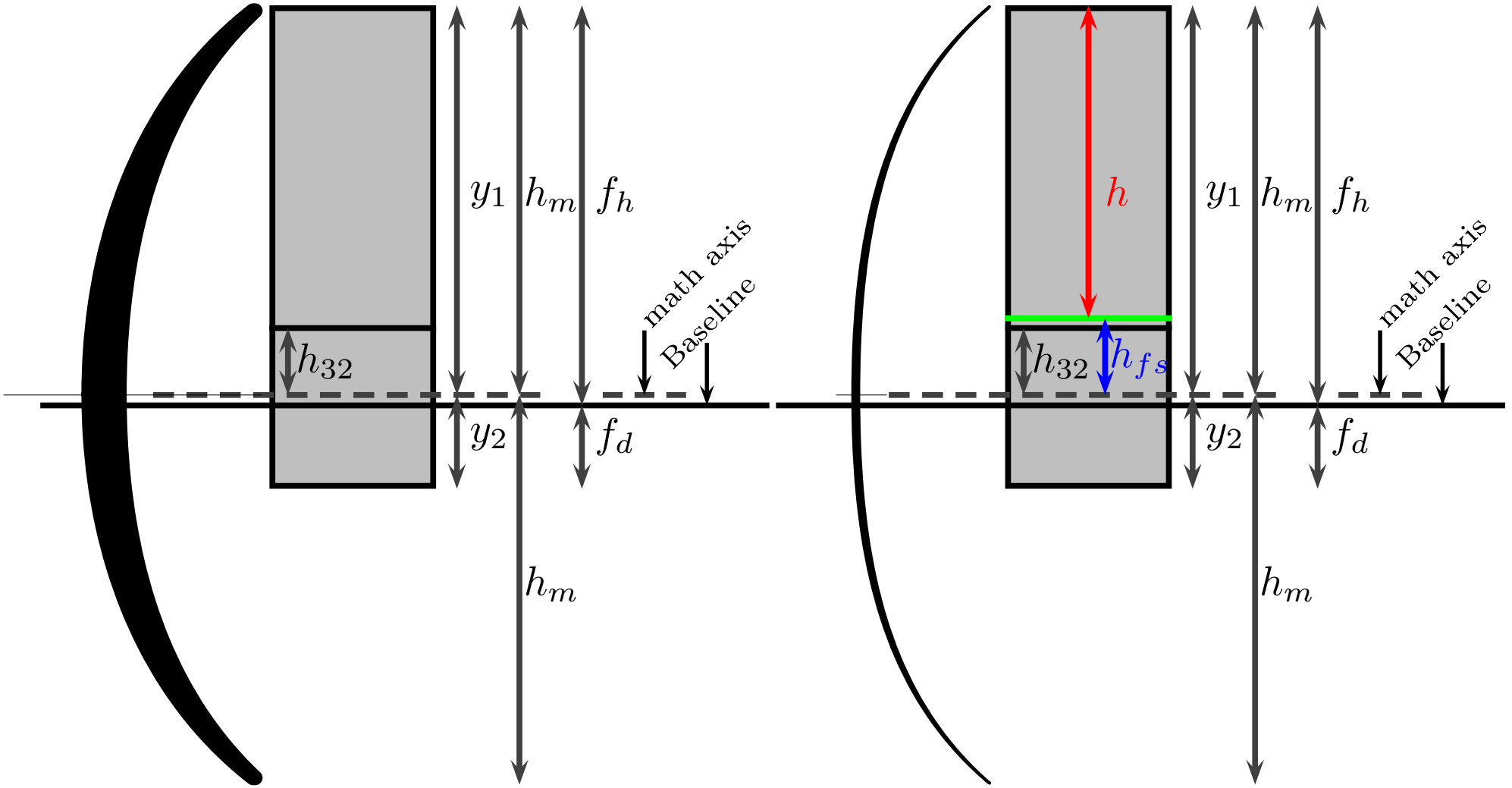
$$fs = \frac{1003.75 \times h_m}{h_{1000}}$$



Scaling - $h_m > h_{32}$



Scaling - $h_m > h_{32}$



Scaling - $h_m > h_{32}$

$$h^p_{max} = \frac{32700 \times 32}{1000} = 1190.4$$

Scaling - $h_m > h_{32}$

$$\boxed{h^p_{max} = \frac{32700 \times 32}{1000} = 1190.4}$$

$$h_{max} = 1.00375 \times h^p_{max}\mathrm{pt}$$

Scaling - $h_m > h_{32}$

$$\boxed{h^p_{max} = \frac{32700 \times 32}{1000} = 1190.4}$$

$$h_{max} = 1.00375 \times h^p_{max}\mathrm{pt}$$

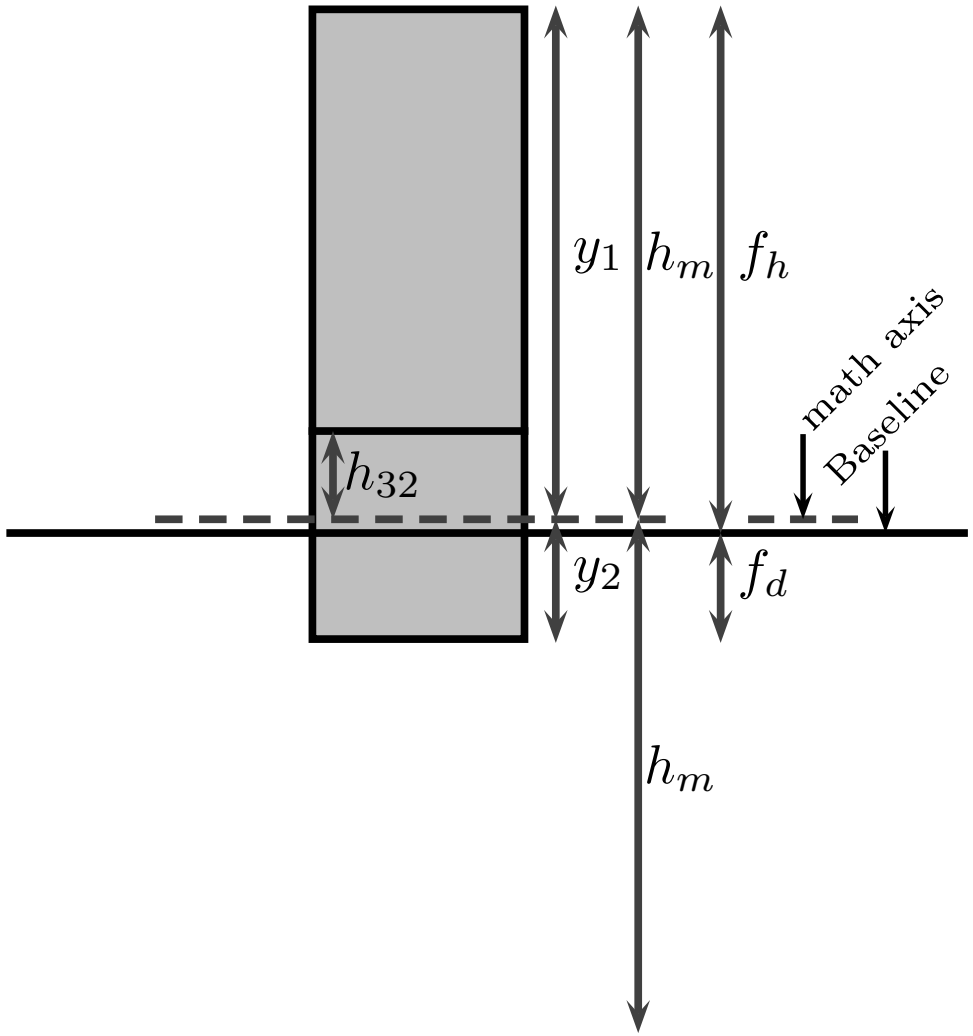
$$\boxed{h_{max} = 1194.864\mathrm{pt}}$$

Scaling - $h_m > h_{32}$

$$\boxed{e\left(h_m\right) = c_1 h_m + c_0}$$

such that:

- $e\left(h_{32}\right) = e_{32}$
- $e\left(h_{max}\right) = \lambda e_{32}$
- e_{32}: thickness of the dynamic symbol in body 32
- λ: a scaling factor (3.236)

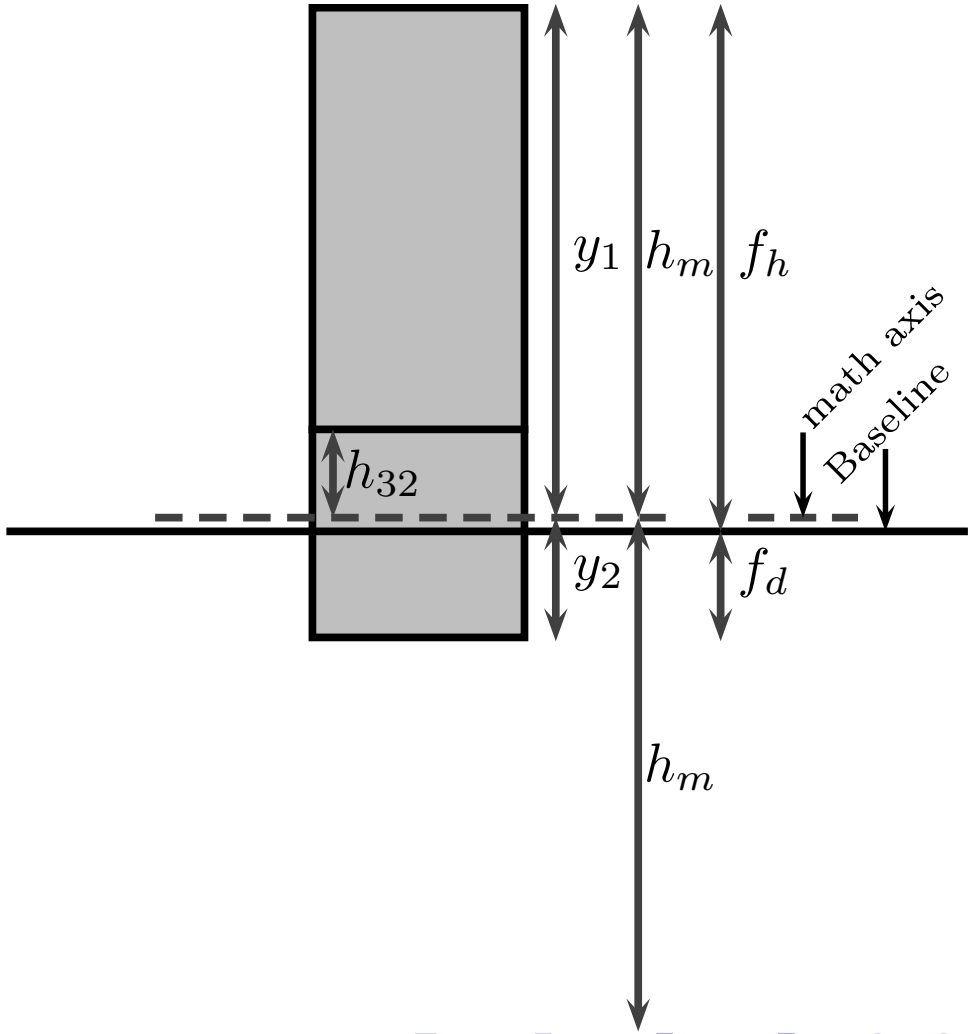


Scaling - $h_m > h_{32}$

$$\boxed{fs = \frac{1003.75}{e_{1000}} e}$$

Scaling - $h_m > h_{32}$

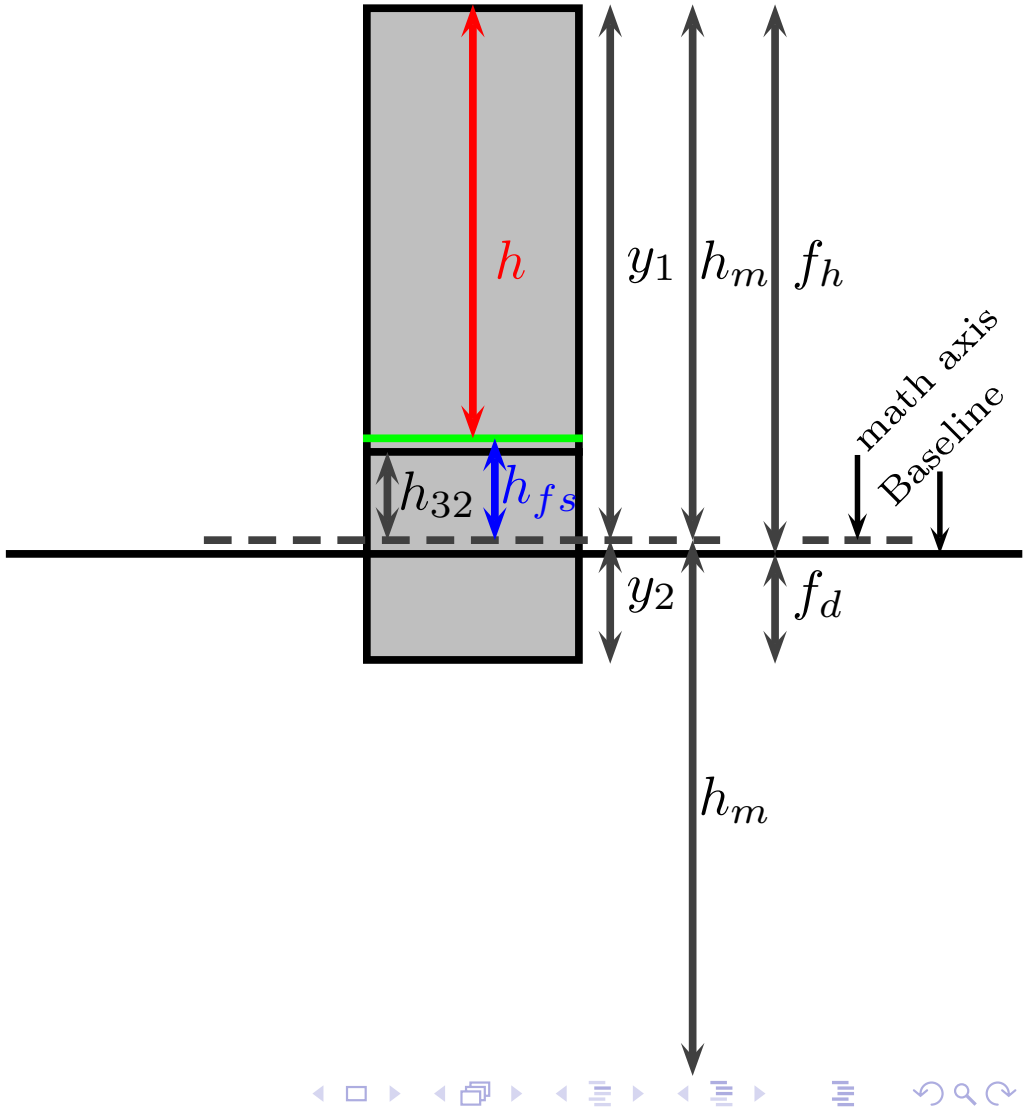
$$fs = \frac{1003.75}{e_{1000}} e$$



Scaling - $h_m > h_{32}$

$$fs = \frac{1003.75}{e_{1000}} e$$

$$h_{fs} = \frac{h_{1000}}{1003.75} fs$$

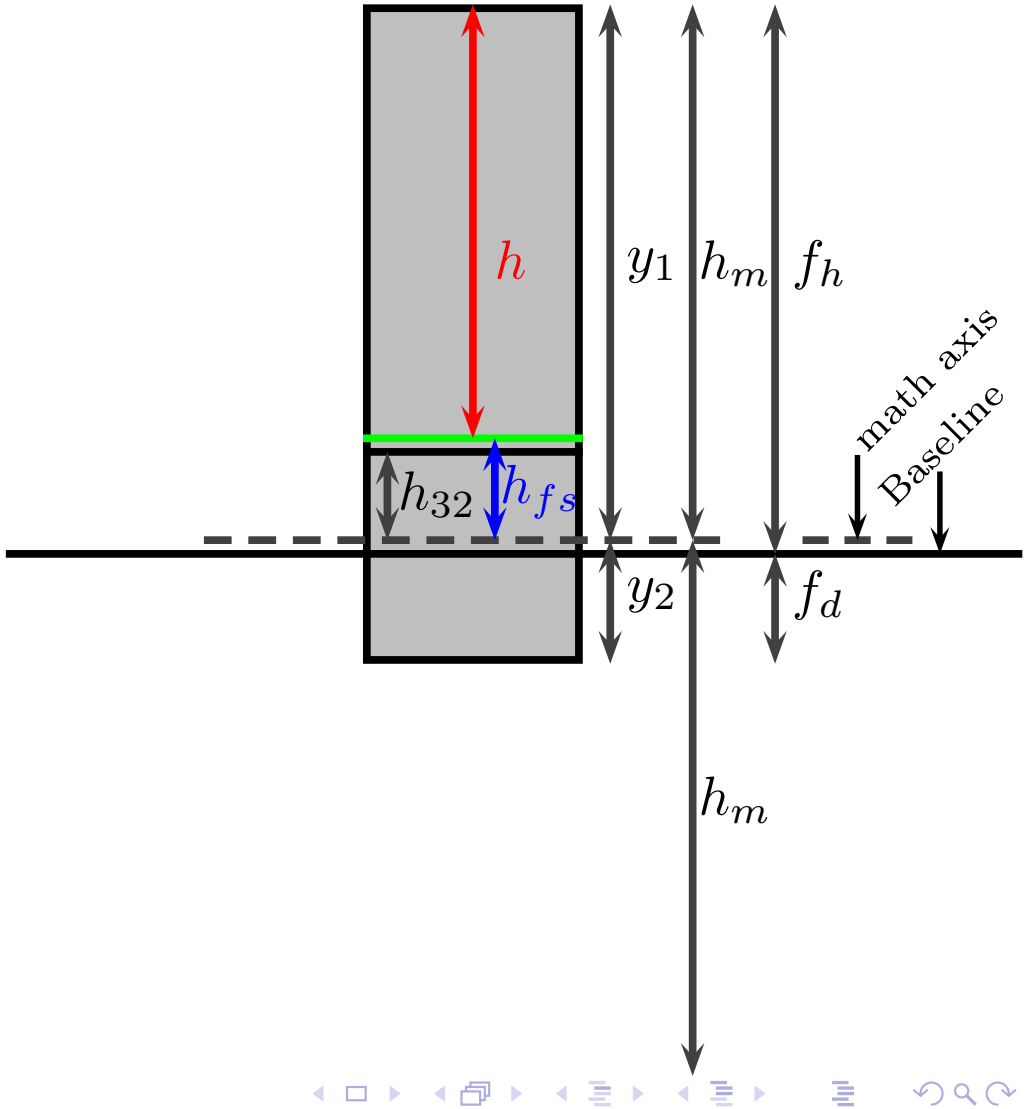


Scaling - $h_m > h_{32}$

$$fs = \frac{1003.75}{e_{1000}} e$$

$$h_{fs} = \frac{h_{1000}}{1003.75} fs$$

$$h = h_m - h_{fs}$$



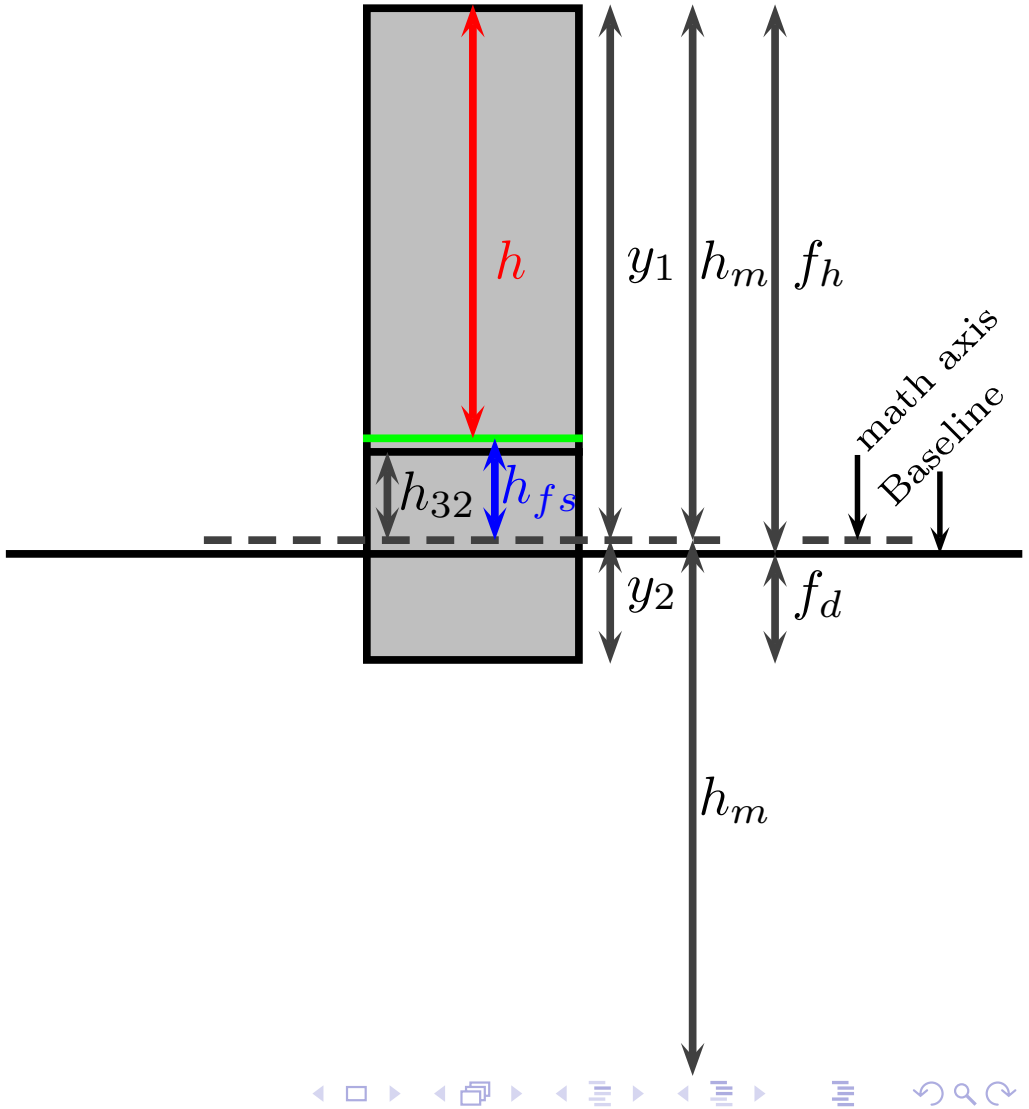
Scaling - $h_m > h_{32}$

$$fs = \frac{1003.75}{e_{1000}} e$$

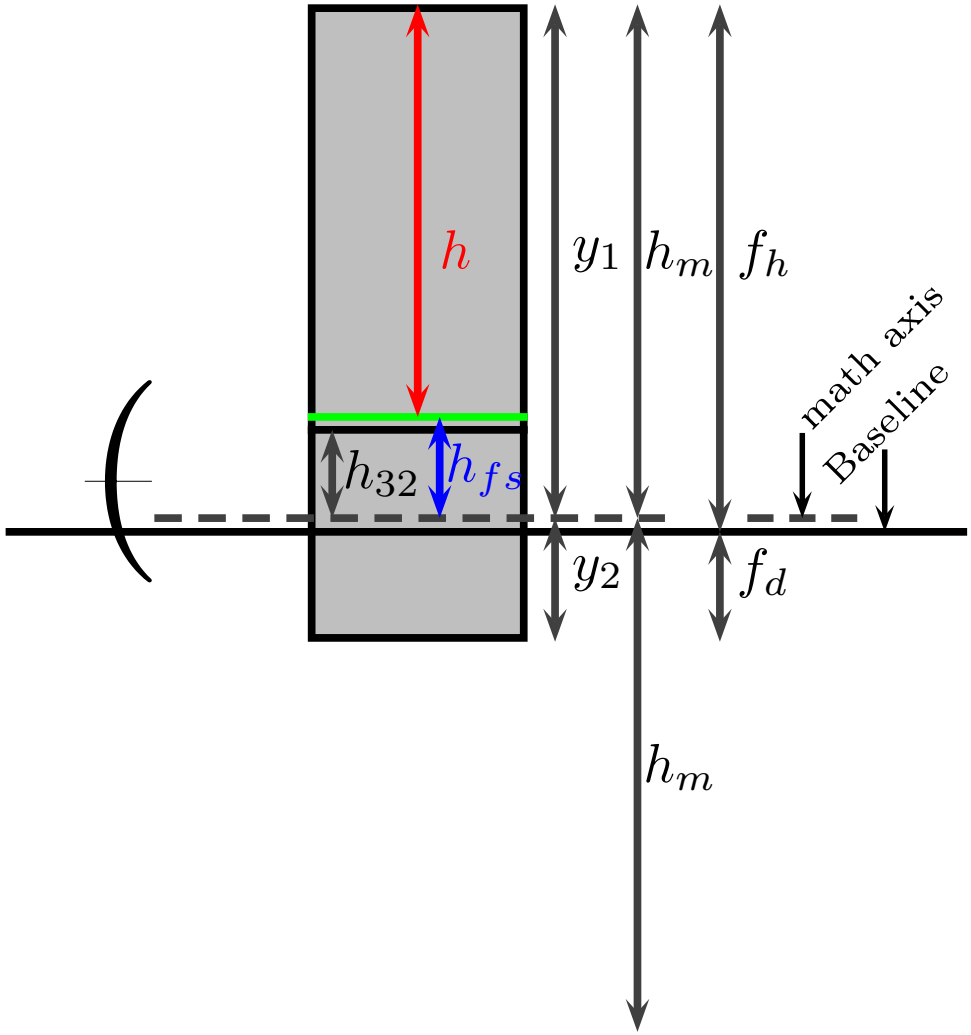
$$h_{fs} = \frac{h_{1000}}{1003.75} fs$$

$$h = h_m - h_{fs}$$

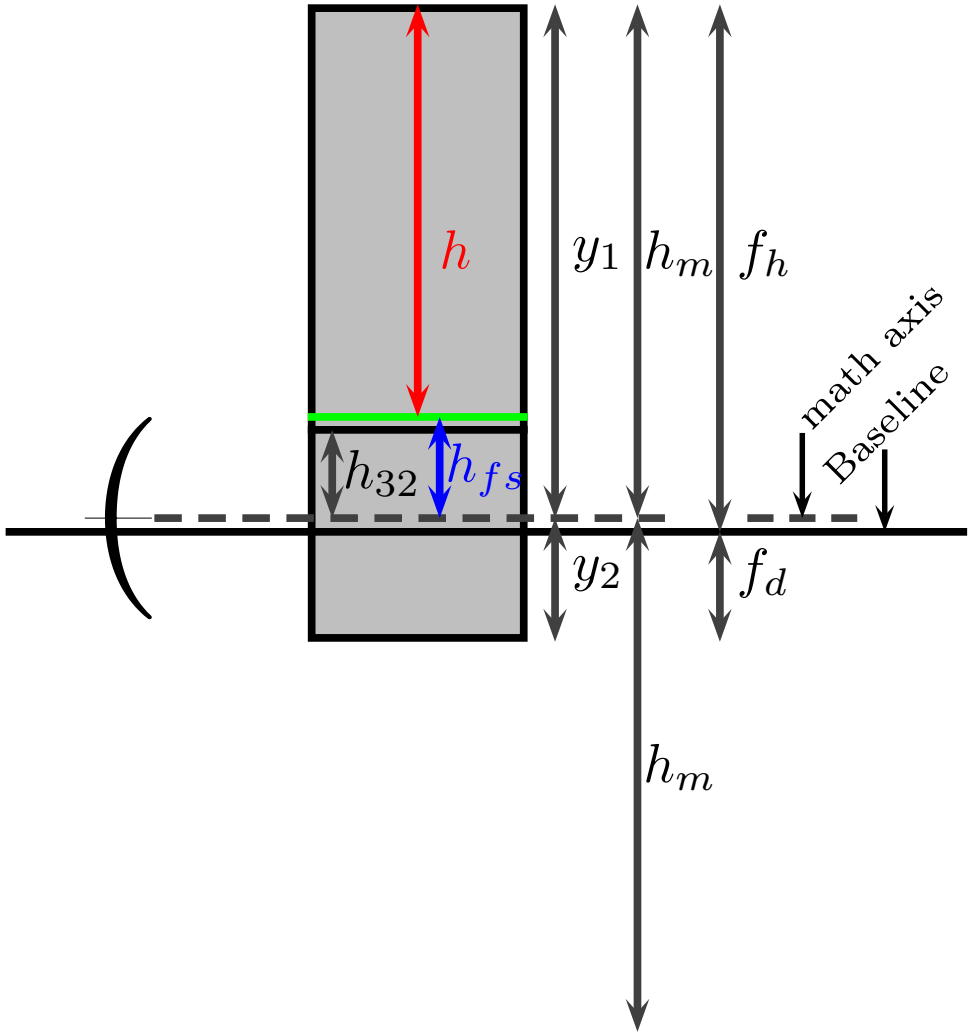
w=function(h)



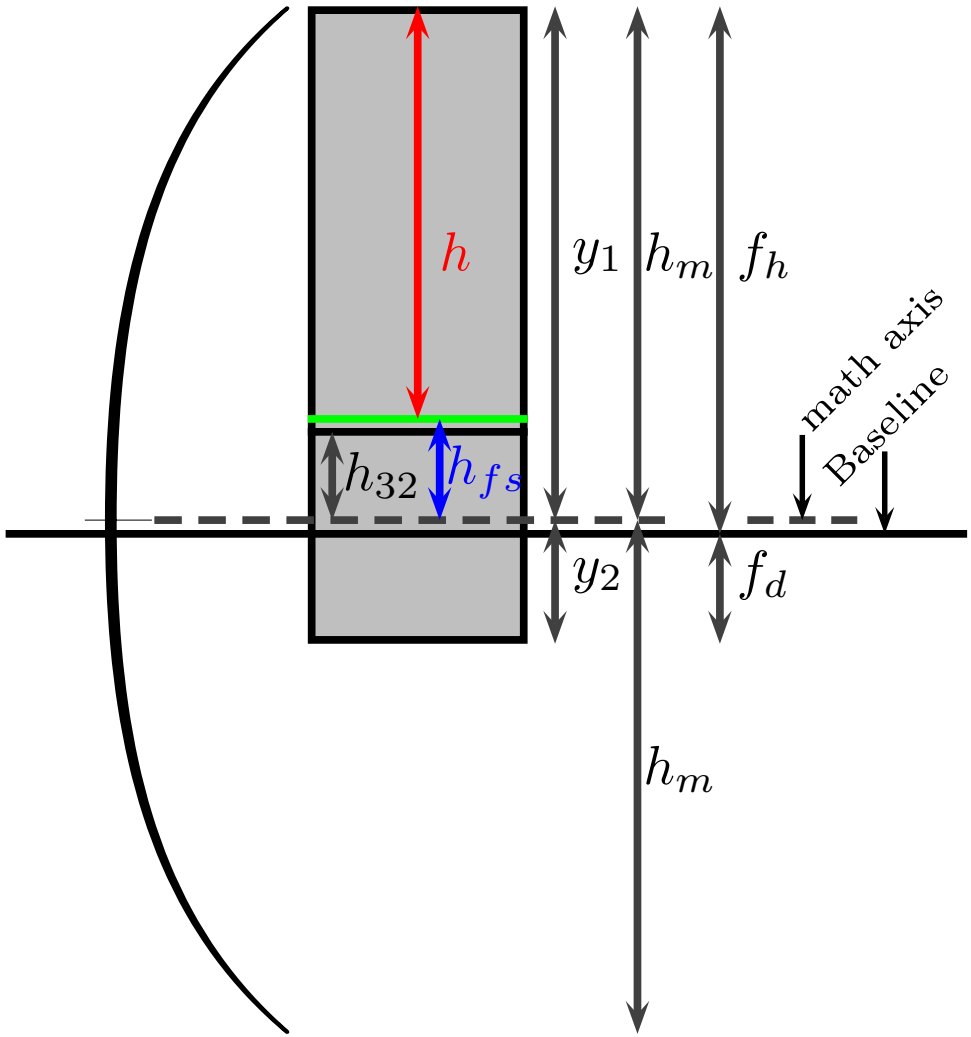
Scaling - $h_m > h_{32}$



Scaling - $h_m > h_{32}$



Scaling - $h_m > h_{32}$



- dynMath: Simple TEX source File
- dynMath: Mini-package
- dynMath for all(TEX): limited and slow
- dynMath for LuaTEX (dviluaTEX and dviluaLATEX): good

- Conclusions:
  - Support of dynamic mathematical symbols with respect to Optical Scaling is feasible.
  - The feasibility is proved with Parentheses and Braces which are adequate representing samples.
- Perspectives:
  - Support of all dynamic mathematical symbols
  - Improvement of quality of symbols (Study of optical scaling considering the artistic viewpoint)