Using MathJax and its Accessibility Features

Volker Sorge, Peter Krautzberger

MathJax

AHG 2016, Denver, November 16 2016

What is MathJax?

- MathJax is a JavaScript library for rendering Mathematics in all browsers
- Can take LaTeX, AsciiMath, and MathML as input
- Generates browser output, e.g. HTML/CSS, SVG
- Standard Maths rendering solution for: stackexchange, wordpress blogs, mediawiki, etc.

MathJax is the de facto rendering solution of (nearly) all Mathematics on the web (35 million unique daily rendering requests via CDN)
http://www.mathjax.org

- Use it directly from CDN
- Configure according to the need of your web document
- Local installations possible
- Detailed documentation available at:
  http://docs.mathjax.org
- Large user community and support

- Load directly from Content Distribution Network
  - Include single line script tag into web document
  - Example with broad, standard configuration

```
<script
 src='https://cdn.mathjax.org/mathjax/latest/MathJax.js?
 config=TeX-AMS-MML_HTMLorMML'></script>
```

- Local configurations to customise for your web content
  - Allows for fine-grained control of MathJax's behaviour
  - Needs to be added BEFORE the CDN call
  - Example for including inline LaTeX formulas:

```
<script type="text/x-mathjax-config">
MathJax.Hub.Config({
  tex2jax: {
    inlineMath: [['$','$'],['\\(','\\)']]
  },
});
</script>
```

- Inbuilt and optional accessibility features
- Selectable in context menu since MathJax v2.7

More details at:
https://www.mathjax.org/
mathjax-accessibility-extensions-v1-now-available/

MathJax's AT Features

- Magnification
- Responsive Equations and Abstraction
- Highlighting
- Interactive Exploration
- Speech Generation

Magnification

- Zoom feature for single math expression
- Magnification up to 500%
- Option selectable in context menu
- Customisable trigger via mouse and keyboard actions

Responsive Equations

- Responsive design enhances reflow and readability of math documents
- Automatic reflow for simplifying layout, adapting to form factor of display and magnification
- Intelligent linebreaking by exploiting semantic enrichment
  - Don't break in the middle of an expression
- Chunking: Abstracting over large elements
  - collapsing mathematically meaningful sub-expressions

Interactive Exploration

- "Walkers" allow to interactively dive into mathematical expression
- Synchronised highlighting together with aural rendering
- Use <shift><space> to explore expressions
- Simple navigation model using arrow keys
- Different types of walkers: syntactic, semantic
- Interactive collapse and expansion of sub-expressions

Aural Rendering and Highlighting

- Speech strings are computed with Speech Rule Engine (SRE)
- Currently uses the MathSpeak rules: verbose, brief, superbrief
  - special summarisations for collapsed parts
  - Other rule sets and localisations in the future
- WAI-ARIA and CSS to implement interactive exploration
  - Speech output by updating ARIA live regions
  - Colour/contrast changes by rewriting CSS properties
- Speech strings can be precomputed or generated on the fly
- Works for all renderers MathJax provides

Exercise

1. Build a web document for the quadratic formula:

   ```
   \[
       x = \frac{-b \pm \sqrt{b^2-4ac}}{2a}
   \]
   ```

2. Include basic MathJax configuration, e.g. take it from
   https://www.mathjax.org/#docs

3. Experiment with the MathJax accessibility extension
   1. Load it from sub menu
   2. Use <shift><space> to explore expressions
   3. Experiment with NVDA screen reader

Exercise Advanced

After loading the explorer extension, you will see that subtitles are still greyed out in the sub-menu.

4. Switch sub-titles on programmatically in your page
   1. Add a configuration option for MathJax
   2. Make sure to add it BEFORE the call to CDN

   ```
   menuSettings: {
     'Assistive-subtitle': true
   }
   ```

For other demonstrations and experimental tools follow the links at
`https://github.com/mathjax/MathJax-a11y`