
Workshop: Advanced JSXGraph I

2. International JSXGraph Conference

Alfred Wassermann



05-10-2021

Contents

Introduction	3
Preliminaries	3
Include JSXGraph	3
JSXGraph skeleton page	3
JSXGraph skeleton page for developer	3
JSXGraph - input from the 2020 conference	4
TypeScript support via <code>index.d.ts</code>	4
Responsive design	4
JSXGraph and accessibility	6
More new features from 1.2.0 to 1.3.2	7
<code>point.isOn(tol)</code> : exact method for <i>incidence</i> of points.	7
Clipping: <code>curveintersection</code> , <code>curvedifference</code> , <code>curveunion</code>	8
Arrow <i>type 7</i>	8

Introduction

The plan of this workshop is to summarize what happened since October 2020 and - additionally - learn some lesser known features of JSXGraph.

Preliminaries

Include JSXGraph

JSXGraph skeleton page

- JSXGraph in production: include `jsxgraphcore.js`

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>JSXGraph template</title>
    <meta content="text/html; charset=utf-8" http-equiv="Content-Type">
    <link href="https://cdn.jsdelivr.net/npm/jsxgraph/distrib/jsxgraph.css" rel="
      stylesheet" type="text/css" />
    <script src="https://cdn.jsdelivr.net/npm/jsxgraph/distrib/jsxgraphcore.js"
      type="text/javascript" charset="UTF-8"></script>

    <!-- The next line is optional: MathJax -->
    <script src="https://cdn.jsdelivr.net/npm/mathjax@3/es5/tex-ctml.js" id="
      MathJax-script" async></script>
  </head>
  <body>

  <div id="jxgbox" class="jxgbox" style="width:500px; height:200px;"></div>

  <script>
    var board = JXG.JSXGraph.initBoard('jxgbox', {boundingbox: [-5, 2, 5, -2]});
  </script>

  </body>
</html>
```

- See JSXGraph handbook (in development): <https://ipesek.github.io/jsxgraphbook/> for an introduction.

JSXGraph skeleton page for developer

- Developer mode: include `jsxgraphsrc.js`

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>JSXGraph template</title>
    <meta content="text/html; charset=utf-8" http-equiv="Content-Type">
    <link href="https://cdn.jsdelivr.net/npm/jsxgraph/distrib/jsxgraph.css" rel="
      stylesheet" type="text/css" />
    <script src="https://cdn.jsdelivr.net/npm/jsxgraph/distrib/jsxgraphsrc.js" type
      ="text/javascript" charset="UTF-8"></script>

    <!-- The next line is optional: MathJax -->
    <script src="https://cdn.jsdelivr.net/npm/mathjax@3/es5/tex-ctml.js" id="
      MathJax-script" async></script>
  </head>
  <body>

    <div id="jxgbox" class="jxgbox" style="width:500px; height:200px;"></div>

    <script>
      var board = JXG.JSXGraph.initBoard('jxgbox', {boundingbox: [-5, 2, 5, -2]});
    </script>

  </body>
</html>
```

JSXGraph - input from the 2020 conference

TypeScript support via `index.d.ts`

- Contributed by David Holmes from STEMStudio <https://stemcstudio.com/standalone>
- Not fully complete. Please, send pull requests if something is missing.

Responsive design

See talk by Murray Bourne at the [JSXGraph conference 2020](#)

Challenge: How can a JSXGraph construction react to varying screen sizes, i.e. in case of orientation changes of a mobile phone or tablet.

1. CSS attribute `aspect-ratio`

- <https://jsfiddle.net/ow0zkryL/>
- CSS: `max-width:100%; aspect-ratio: 3/2;`
- `fontUnit: 'vmin'`



New attribute `fontUnit` for texts

```
<div id="jxgbox" class="jxgbox" style="max-width:100%; aspect-ratio: 3/2;"></div>
```

```
const board = JXG.JSXGraph.initBoard('jxgbox', {
  boundingbox: [-5, 5, 5, -5],
  axis:true,
  keepAspectRatio:true
});

var pol = board.create('polygon', [[-3,-3], [3,-3], [1,4]], {
  fillColor: 'yellow',
  vertices: {
    color: 'blue', name: 'A', withLabel: true,
    label: { fontSize: 3, fontUnit: 'vmin' }
  }
});
```

2. The padding-bottom hack

- `aspect-ratio` unsupported on older browsers: <https://caniuse.com/?search=aspect-ratio>
- Presented by Murray Bourne's talk last year: <https://jsxgraph.org/con/program>
- <https://jsfiddle.net/yem1phx5/1/>

```
<div style="max-width: 100%">
  <div id='jxgbox' class='jxgbox' style="height:0; padding-bottom:66%"></div>
</div>
```

3. The padding-bottom hack is compatible with *full screen mode*,

- `board.toFullscreen(divid)`
- see <https://jsxgraph.org/docs/symbols/JXG.Board.html> and go to `toFullscreen`.

4. JSXGraph in tab panels is also working now

- See <https://jsfiddle.net/8khvu409/3/>
- Tab panels in a web page can be easily realized with the CSS pseudo element `:target`

```
<p>
  <a href="#jxgbox1">show box 1</a>
  <a href="#jxgbox2">show box 2</a>
</p>

<div class="items">
  <div id="jxgbox1" class="jxgbox myStyle"></div>
  <div id="jxgbox2" class="jxgbox myStyle"></div>
</div>
```

```
div.items div.jxgbox:not(:target) { display: none; }
div.items div.jxgbox:target { display: block; }

/* Show box 1 on load */
/*
div.items div#jxgbox1 { display: block; }
*/

.myStyle {
  max-width: 80%;
  aspect-ratio: 1/1;
}
```

```
var b1 = (function() {
  var board = JXG.JSXGraph.initBoard('jxgbox1', {
    boundingbox: [-5,5,5,-5],
    axis: true,
    moveTarget: document
  });
  var pol = board.create('polygon', [[-3,-3], [3,-3], [1,4]], { fillColor: 'yellow' });
  return board;
})();

var b2 = (function() {
  var board = JXG.JSXGraph.initBoard('jxgbox2', {
    boundingbox: [-5,5,5,-5],
    keepAspectRatio: true,
    axis: true,
    resize: {enabled: true}
  });
  board.create('polygon', [[-3,-3], [3,-3], [1,4]], { fillColor: 'red' });
  return board;
})();
```



(function(){ ... })(); makes all variable definitions inside {} local. This is helpful if several JSXGraph constructions are on one page.

5. New board attribute `moveTarget`

- <https://jsfiddle.net/o82fmy4e/>
- Controversial: Better handling on small boards but users may “lose” points

JSXGraph and accessibility

ARIA attributes:

- New board attributes `title` and `description`.
- Two new, invisible divs are added which contain these values

- JSXGraph container has new attributes `aria-labelledby` and `aria-describedby`

Keyboard control

- See <https://jsfiddle.net/eLtpg1bn/>
- Move focus with `tab`
- Move position with arrow keys

```
const board = JXG.JSXGraph.initBoard("jxgbox", {boundingbox: [-5,5,5,-5],
  axis: true,
  keyboard: {
    enabled: true,
    dy: 30,
    panShift: true,
    panCtrl: false
  }
});

var p = board.create('point', [-1, 1], { color: 'blue' });
var seg = board.create('segment', [[-2, 0], [-2, 4]]);
var line = board.create('line', [[1, 3], [2, -2]]);
var circ = board.create('circle', [[-4, 3], 1], {tabindex: 0});
```

More new features from 1.2.0 to 1.3.2

JSXGraph 1.2.0 was released in January 2021

`point.isOn(tol)`: exact method for *incidence* of points.

- Avoid “abuse” of `hasPoint`
- <https://jsfiddle.net/95d8rto2/>

```
JXG.Options.precision.mouse = 20;

const board = JXG.JSXGraph.initBoard('jxgbox', {
  boundingbox: [-5, 5, 5, -5],
  axis: true,
  precision: { mouse: 100 }
});

var circle = board.create('circle', [[-3,-3], 2]);
var p = board.create('point', [-3, -1], { snapToGrid: true });
var t1 = board.create('text', [1, -2, () => "isOn " + p.isOn(circle, 0) ], {
  fontSize: 24 });
```

Clipping: curve intersection, curved difference, curve union

- see <http://jsxgraph.org/docs/symbols/CurveIntersection.html>,...

Arrow type 7

- for curves with arrows
- dashed curves work again, <https://jsfiddle.net/by6mpgsk/>

```
// Arrow types: 1 ... 7
var s1 = board.create('segment', [[-2, 3], [2, 3]], { strokeWidth: 10, lastArrow: {
  type: 6 }, dash: 0 });
// var s2 = board.create('segment', [[-2, 1], [2, 1]], { strokeWidth: 10, lastArrow
  : {type: 7 } });

var p1 = board.create('point',[-3, -4], {withLabel: false})
var p2 = board.create('point',[-3, -2], {withLabel: false})
var p3 = board.create('point',[-1, -1], {withLabel: false})
var p4 = board.create('point',[-1, -3], {withLabel: false})
var p5 = board.create('point',[2, -3], {visible: false, withLabel: false})
var p = [p1,p2,p3,p4,p5]

c = board.create('curve', JXG.Math.Numerics.CardinalSpline(p, 1), {
  strokeWidth: 8,
  dash: 0,
  lastArrow: { type: 7 }
});
```