



The Floating Horizon Algorithm

© 1995-2015 Josef Pelikán & Alexander Wilkie
CGG MFF UK Praha

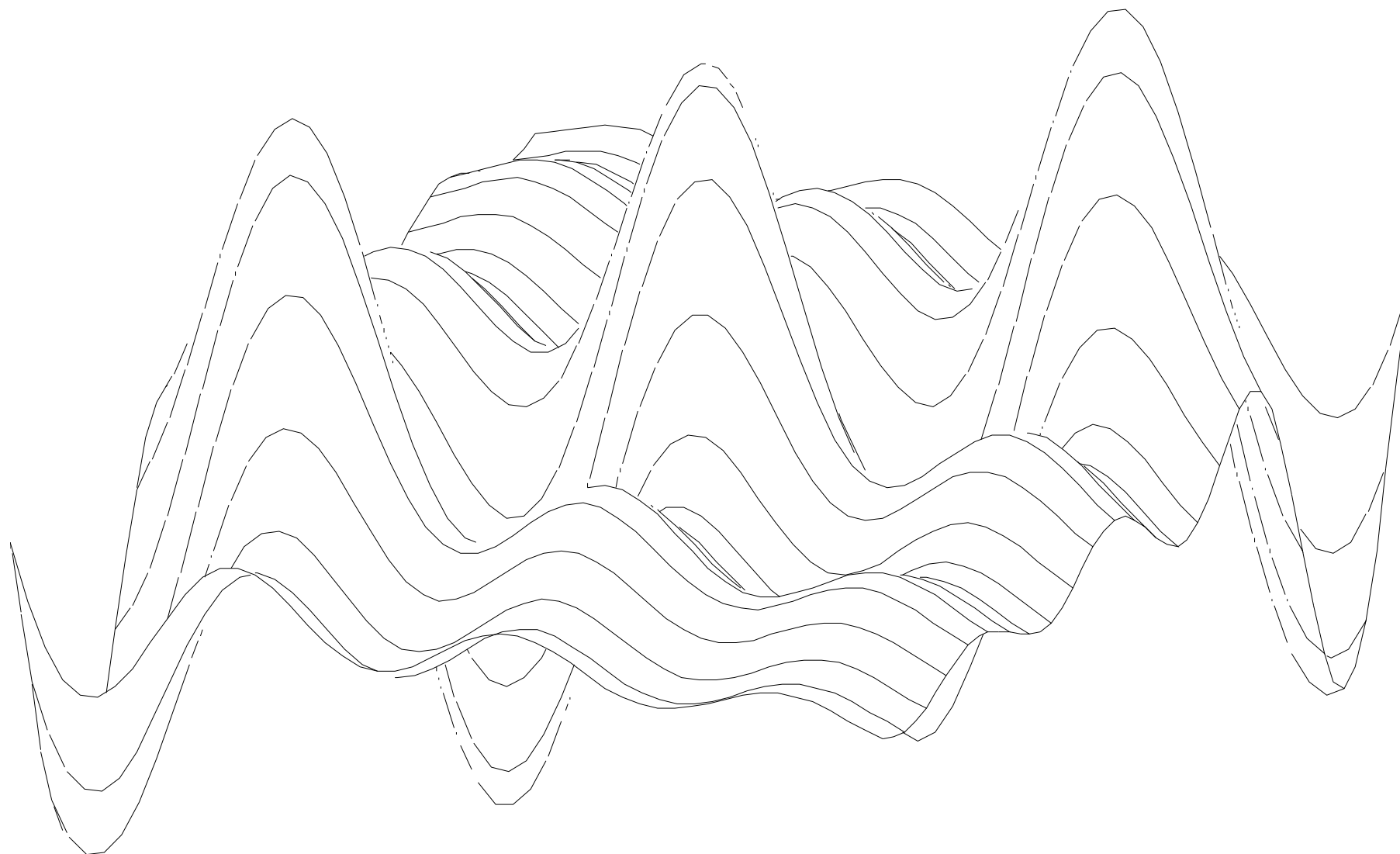
pepca@cgg.mff.cuni.cz

<http://cgg.mff.cuni.cz/~pepca/>

Floating Horizon Algorithm

- Only for **graphs of functions of two variables**
 - Surface **$z = f(x,y)$**
- For some viewpoints of a **central projection** the technique gives incorrect results
- Generates **line drawings**
 - Ideal for vector output on a plotter
- There is also a faster **raster version**
 - There is also a version that fills the area under the curves

Hidden Line Removal



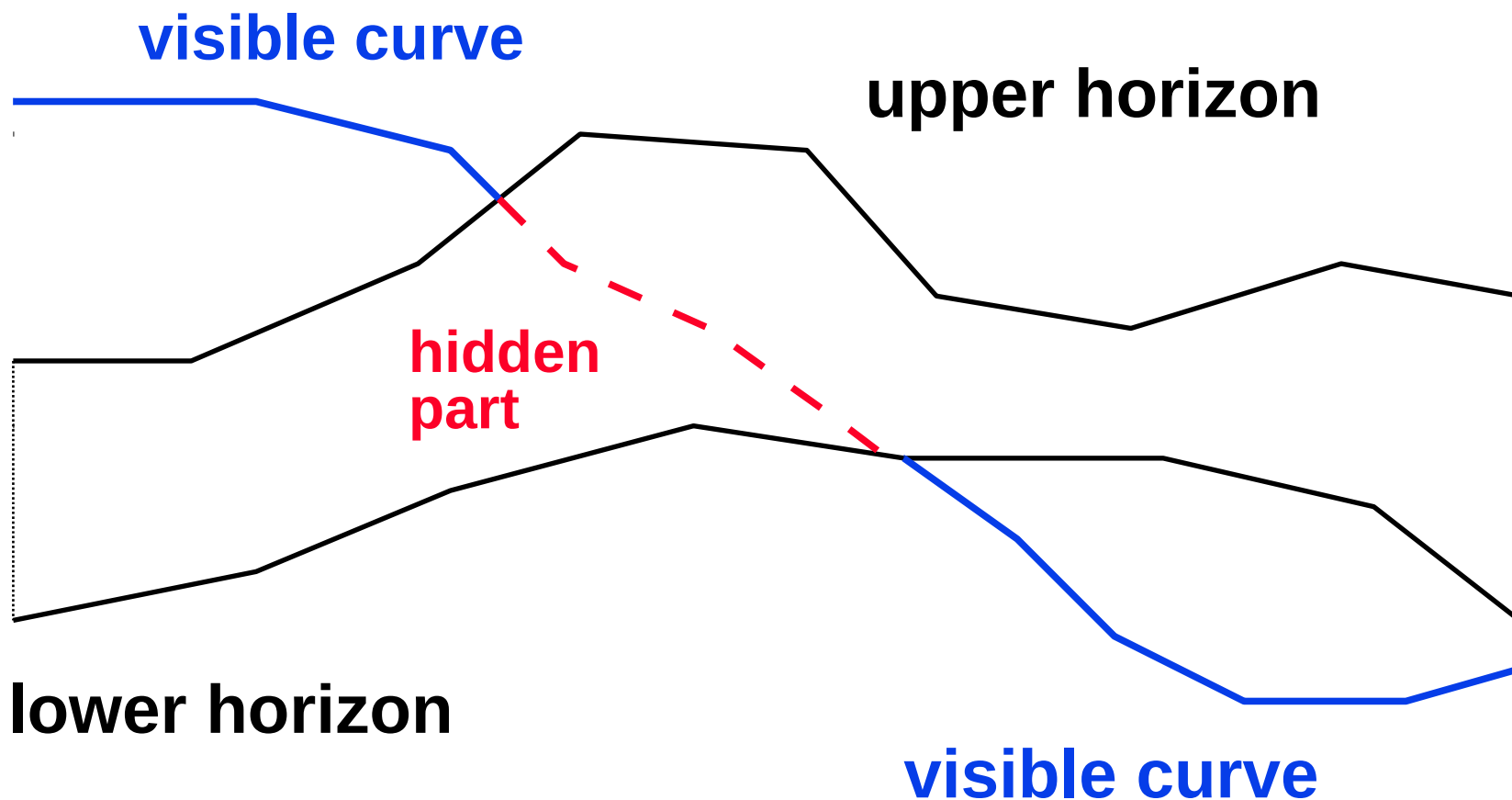


Algorithm Operation

- The surface is drawn **front to back**
 - An object can only be covered by objects that have already been drawn
- We keep the **outline** of the already drawn part of the surface
 - Graph with two unknowns: curves (polylines) – **upper and lower horizon**
- The visible part of the curve must lie **outside the current outline**

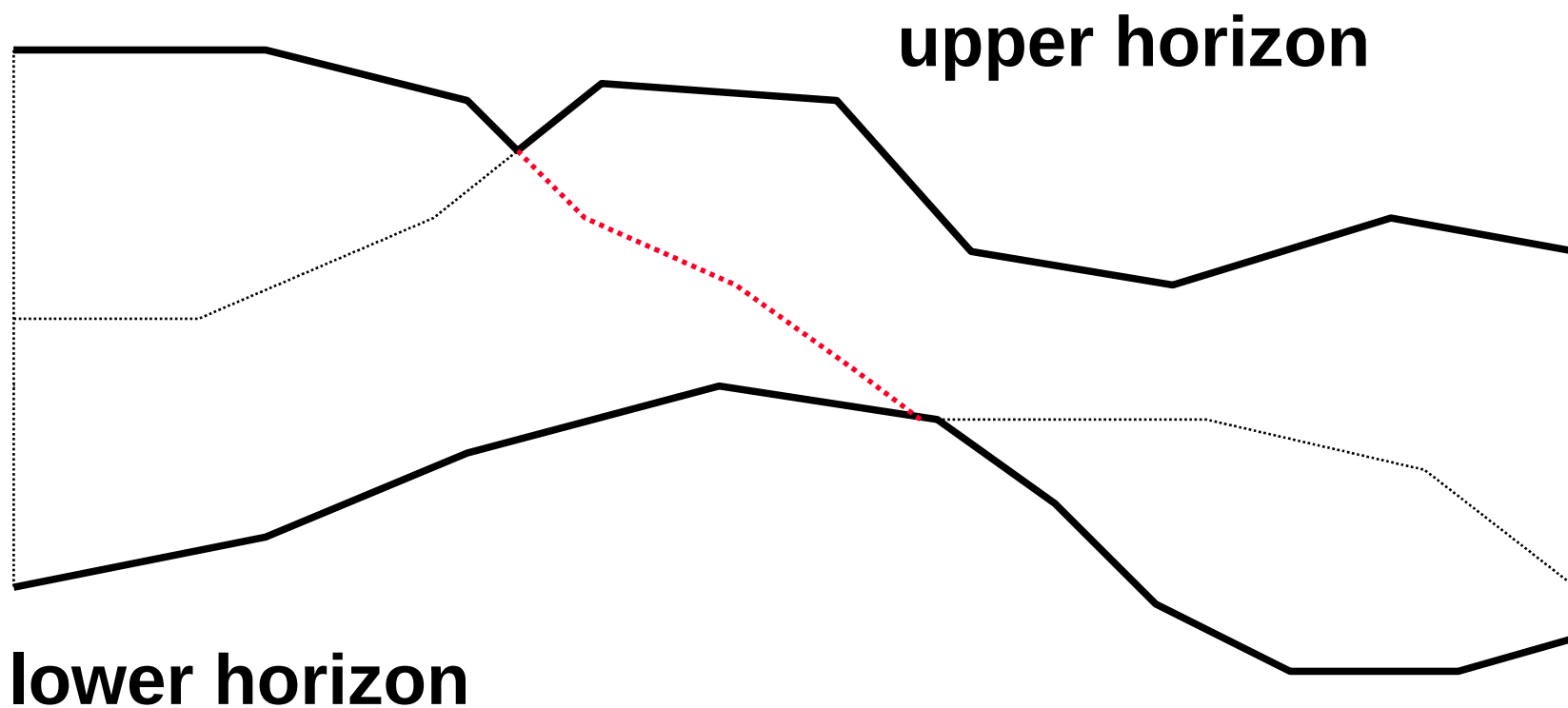


Visibility Computation





Updating the Horizon Curves



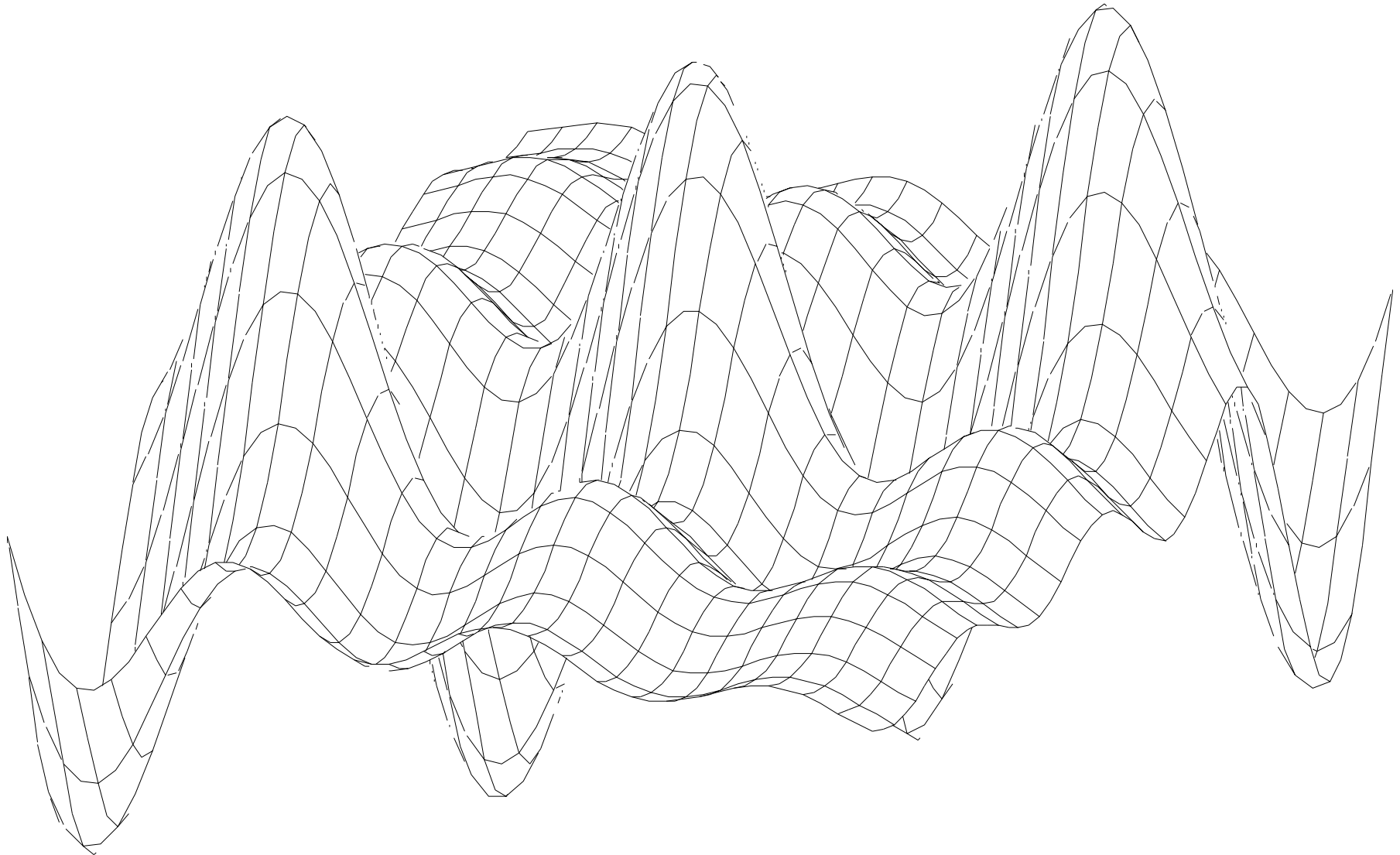


Horizon Implementation

- ① **Polylines:** $[x_1, y_1], [x_2, y_2] \dots [x_n, y_n], x_i < x_{i+1}$
 - Vector output (high precision)
 - Need to compute polyline intersections

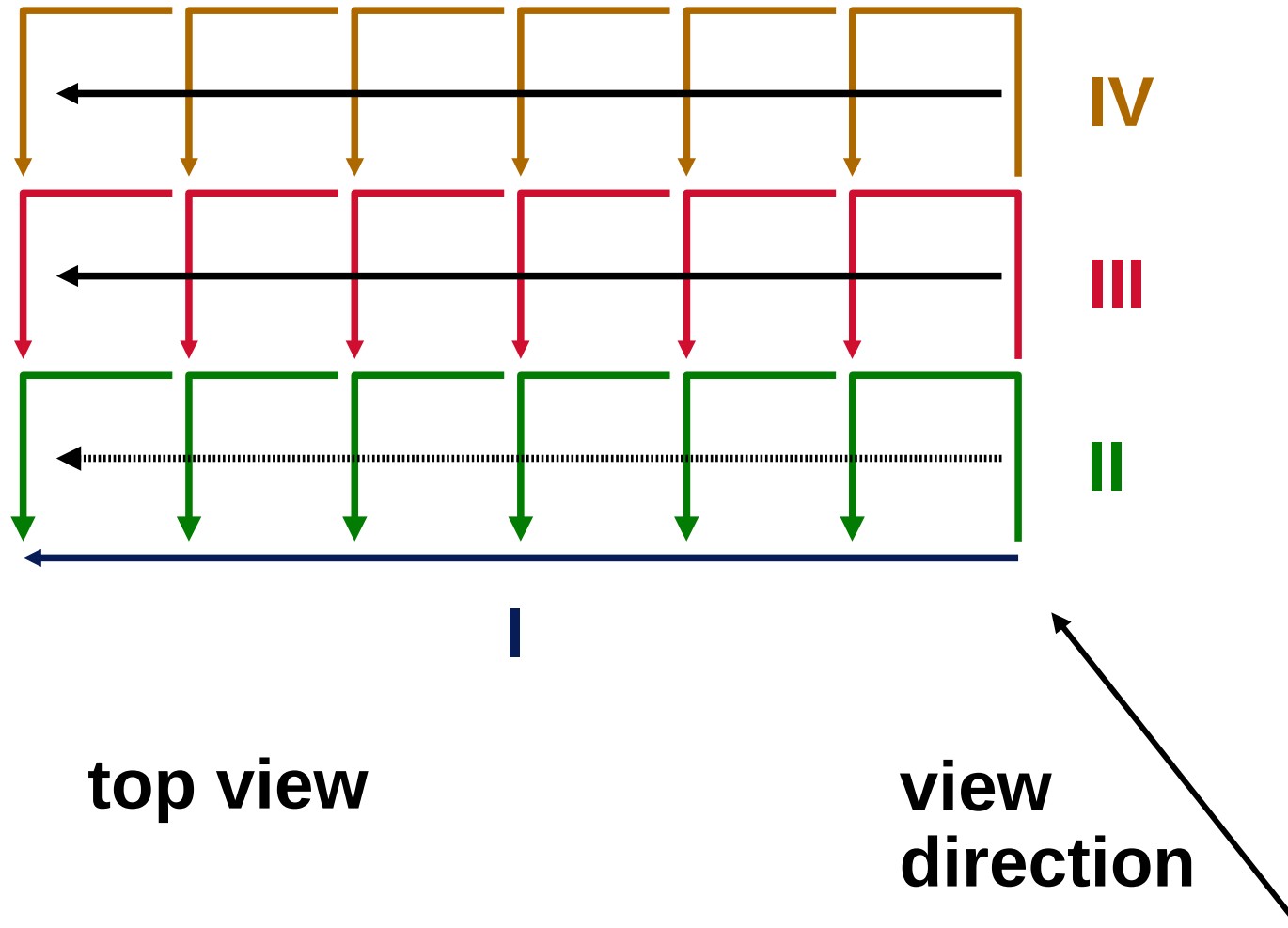
- ② **Via min/max values:** $y_{\min}[i], y_{\max}[i], 0 \leq i < x_{\text{res}}$
 - Raster devices (lower precision)
 - Testing at the pixel level (at lower resolutions, this is efficient)
 - Easy implementation

Two Networks of Surface Curves



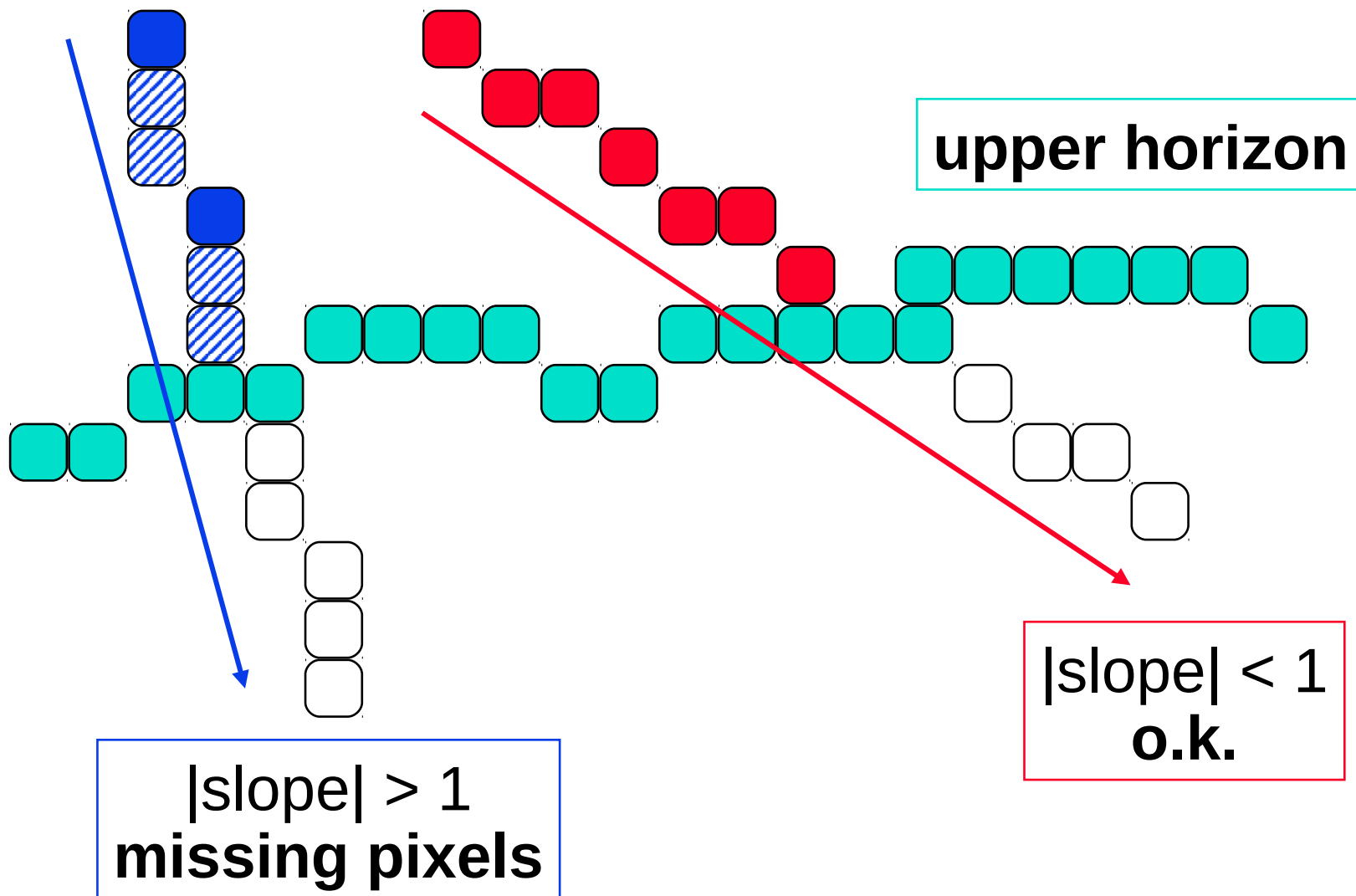


Drawing in z-Order





Steep Lines





Bitmap Algorithm Improvements

- ◆ **„Centrifugal drawing“**
 - Drawing of lines away from the horizon
 - Inefficient for very long segments
- ◆ **Two phase processing** for each segment
 1. Draw the segment (test each pixel)
 2. Update the horizons
- ◆ **Two sets of horizons** (equalised after each segment)
 1. Set is used for trimming the drawn segments
 2. Set is updated based on the drawn segments

End



Further Information:

- **J. Foley, A. van Dam, S. Feiner, J. Hughes:** *Computer Graphics, Principles and Practice*, 651-656
- **Jiří Žára a kol.:** *Počítačová grafika*, principy a algoritmy, 307-311