Moderovacie a renderovacie techniky

František Dráček
dracek1@uniba.sk

7. decembra 2023

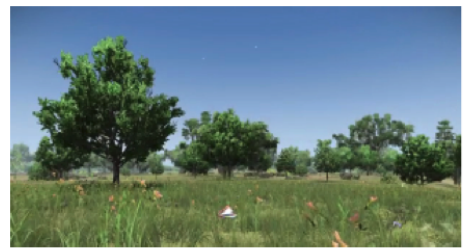
https://github.com/frantisekdracek/Prezentacie/tree/main

Procedural generation

terragen

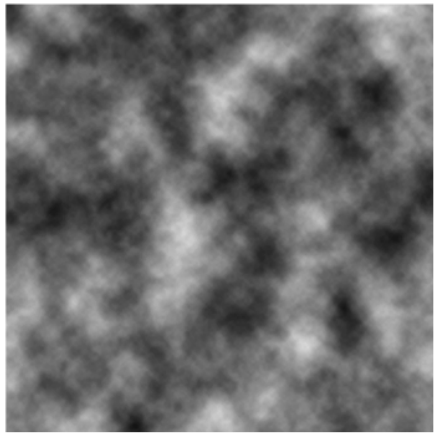
(a)

speedtree

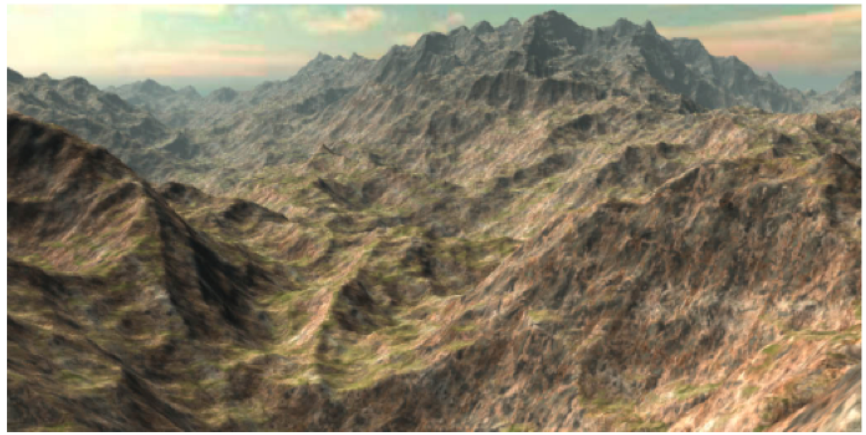


(b)

Obr.: Examples of procedural generation



(a)

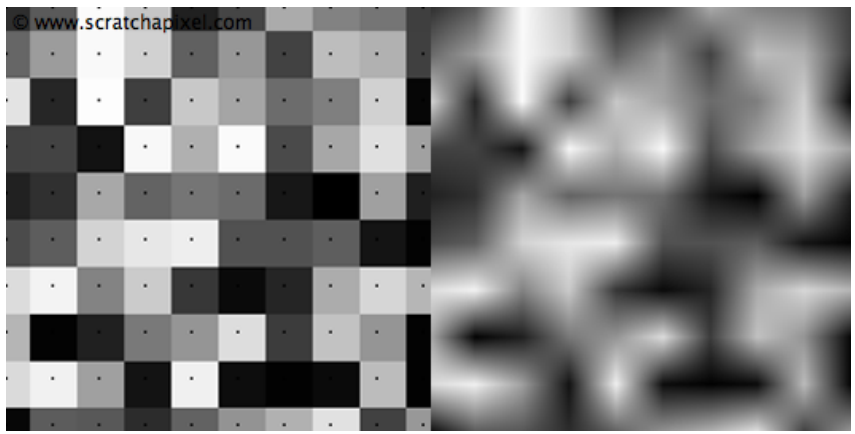


(b)

Obr.: Perlin noise octaves

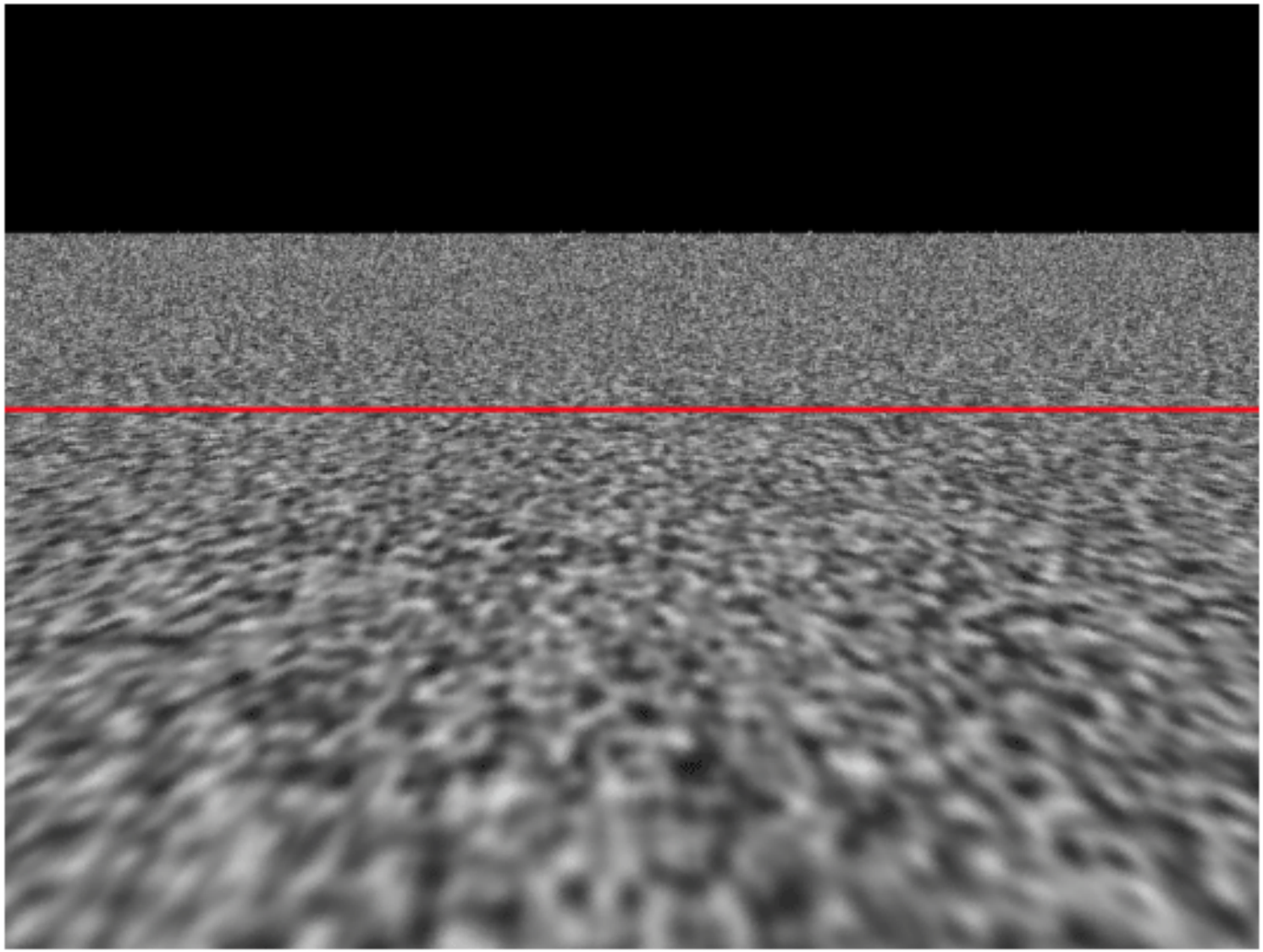
Noise

- developed as alternative to image textures and clean colors
- random patters not sufficient - points that are close are not correlated



Obr.: Noise and linear interpolation

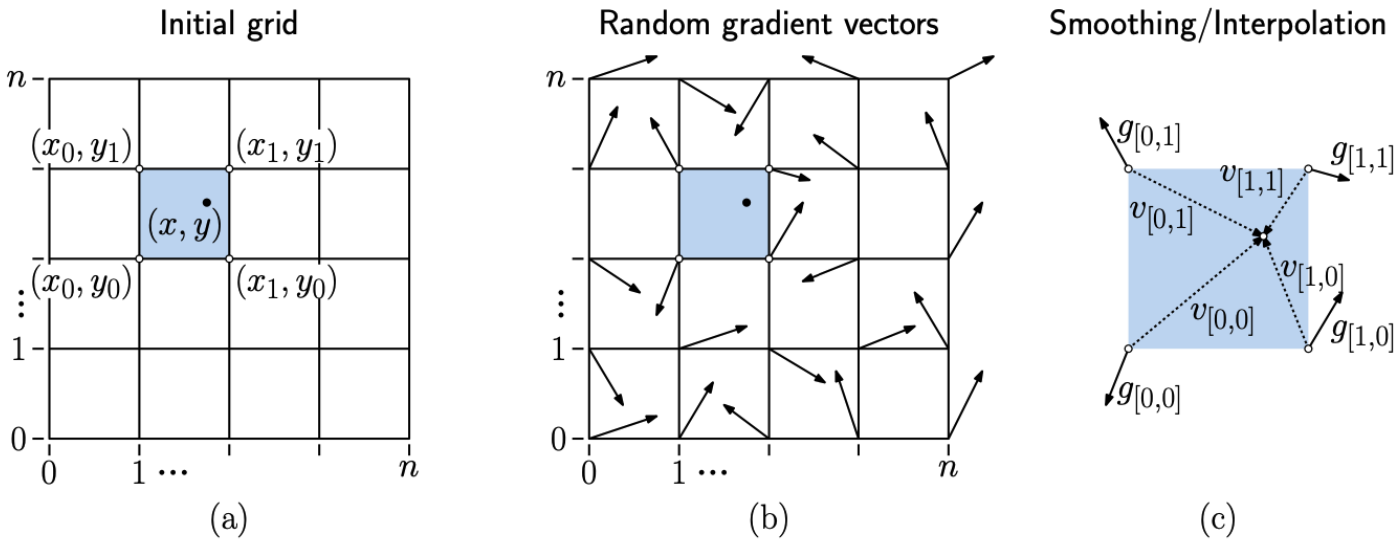
Noise



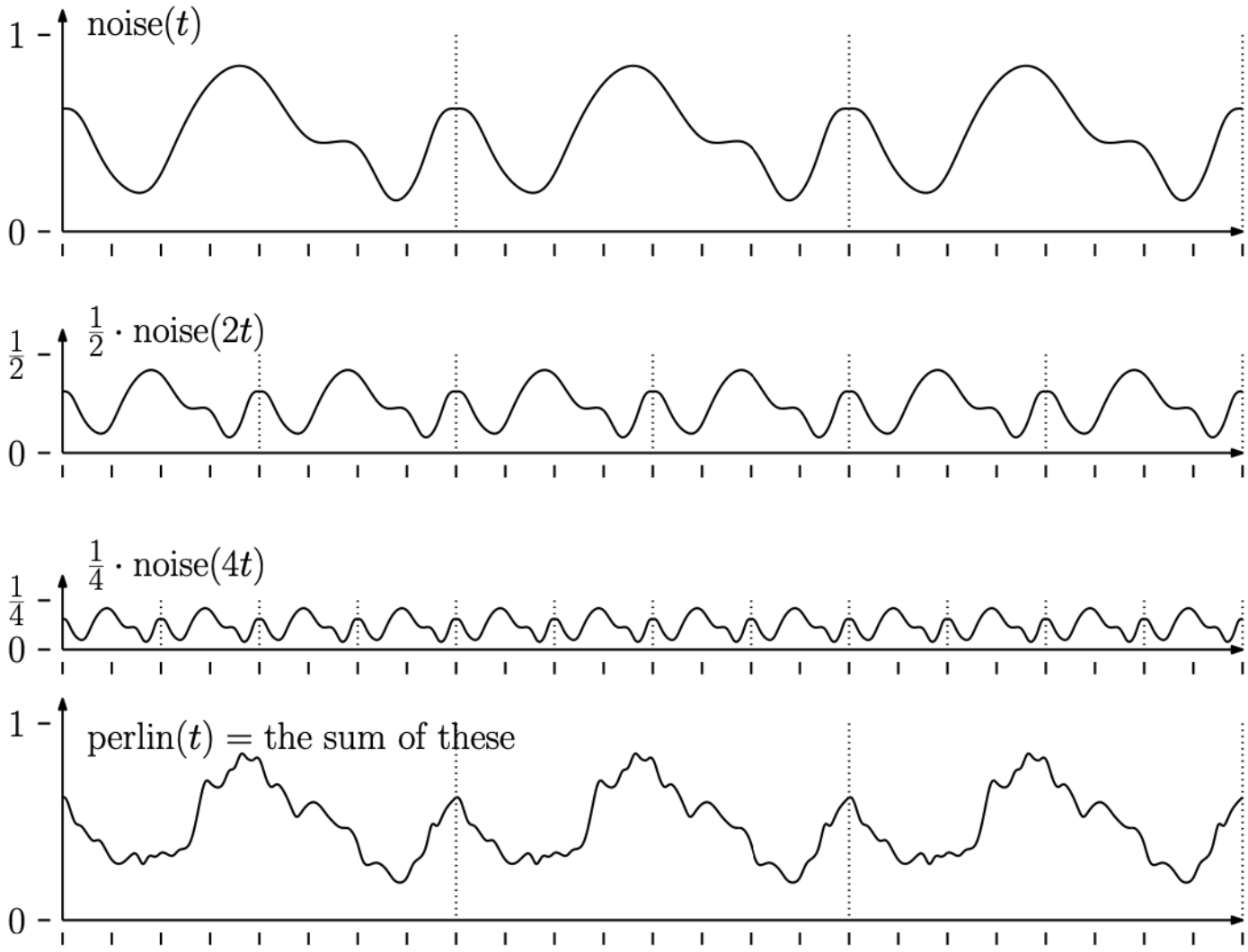
Obr.: Noise with aliasing

Perlin noise

- create lattice
- generate random vector at every corner - gradient $g_{i,j}$
- calculate offset vector from corners to position $v_{i,j}$
- next calculate dot product between corresponding offset and gradient vector $\delta_{i,j} = g_{i,j} \cdot v_{i,j}$
- interpolate between the delta values in cell - fade function
- $\delta_{i,j}$ has zero contributions at corner position and grows with distance, this is undesirable, therefore we need interpolation function
- $\psi(t) = 6t^5 - 15t^4 + 10t^3$
- $\Psi(x, y) = \psi(x)\psi(y)$
- $noise(x, y) =$
  $\Psi(1-x, 1-y)\delta_{0,0} + \Psi(x, 1-y)\delta_{1,0} + \Psi(1-x, y)\delta_{0,1} + \Psi(x, y)\delta_{1,1}$
- $perlin(x, y) = \sum_{i=0}^{k} p^i noise(2^i x, 2^i y)$



Obr.: Generating 2D perlin noise



Obr.: Perlin noise octaves

Thank you!