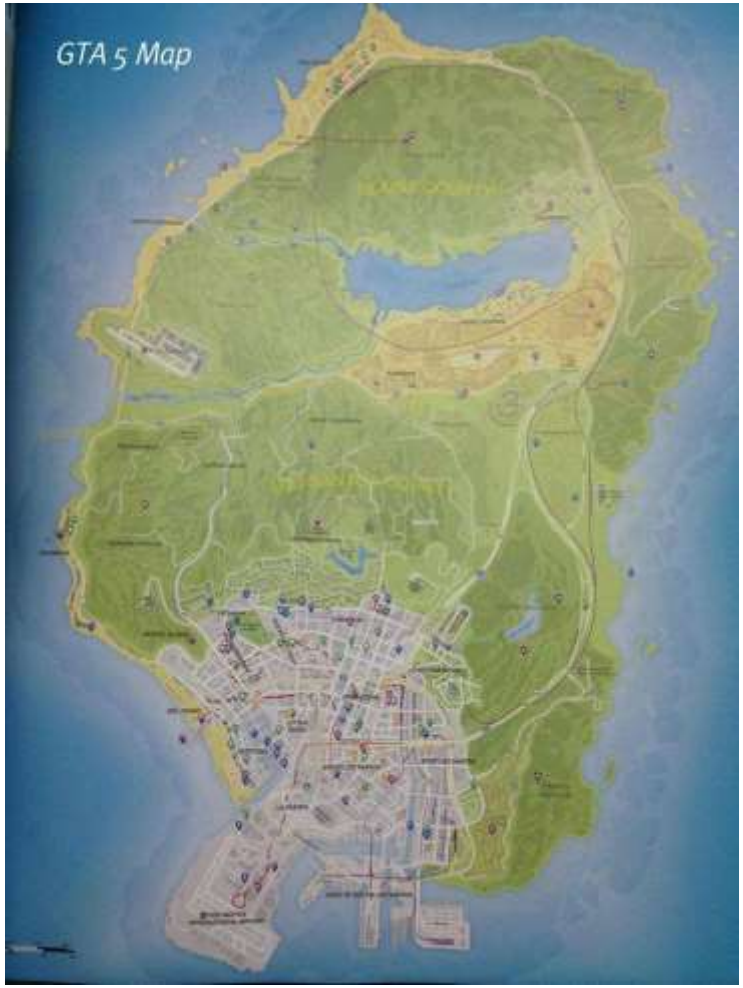
Procedural Content Generation

Metroid, 1986

Metroid Prime III: Corruption, 2007

Content



City of Los Santos,

Grand Theft Auto V, 2013

Content

City of Los Santos,

Grand Theft Auto V, 2013

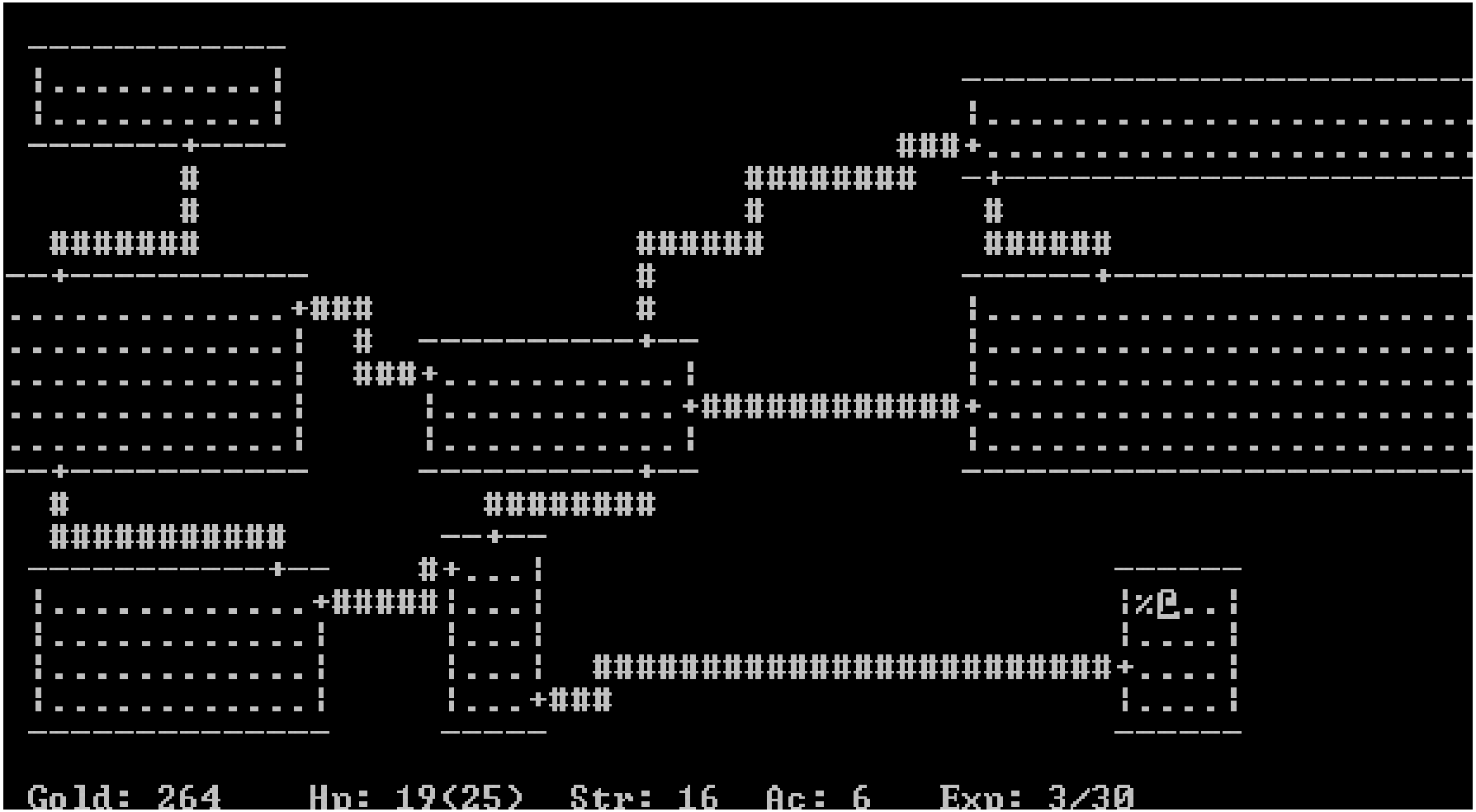
Manhattan (to Scale)

Procedural Content Generation

Procedural content generation (PCG) is the programmatic generation of game content using a random or pseudo-random process that results in an unpredictable range of possible game play spaces.

Source: PCG Wiki (pcg.wikidot.com)

Rogue, 1980

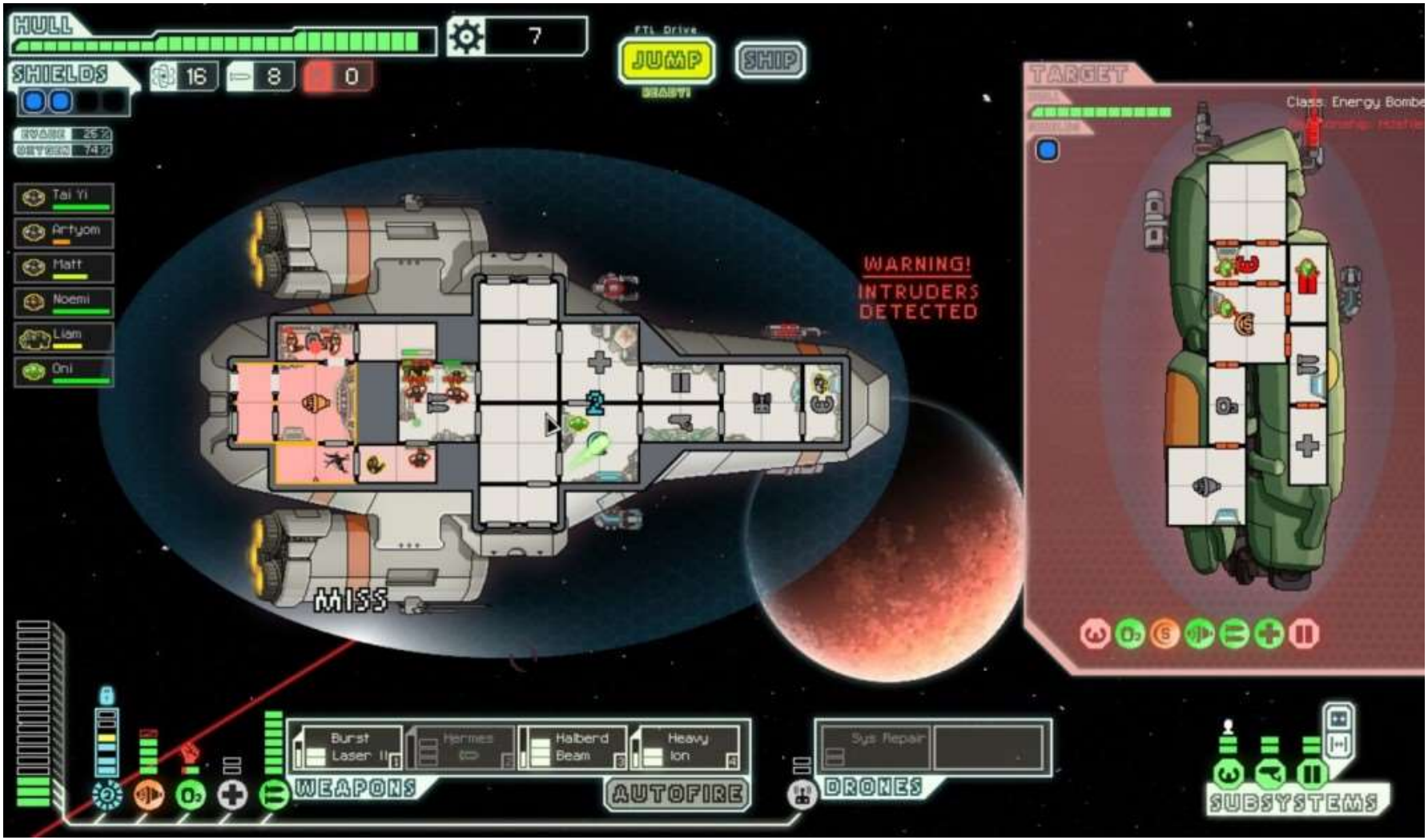


Diablo, 1997

Elite, 1984

Minecraft, 2010

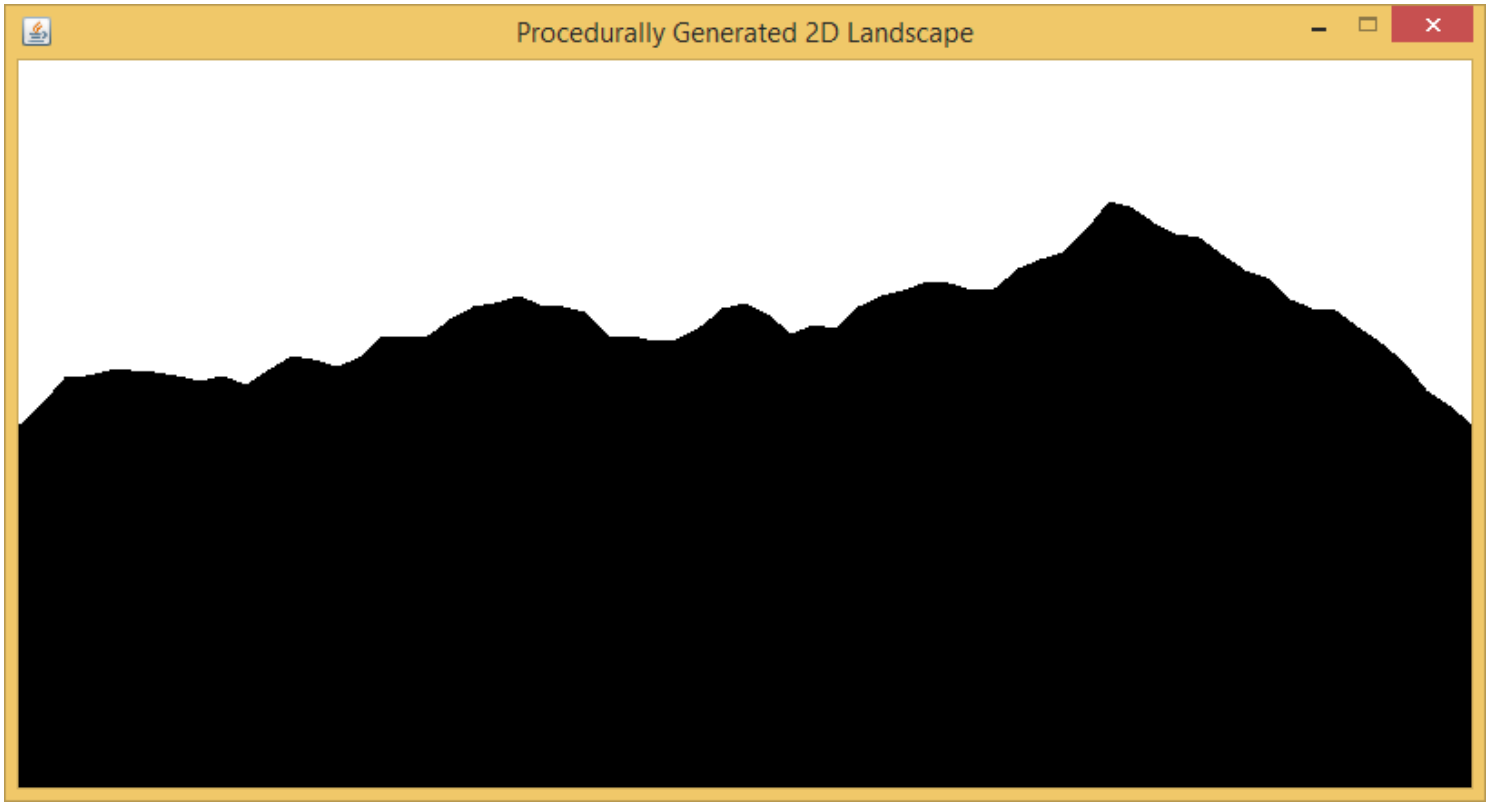
FTL: Faster than Light, 2012



Borderlands II, 2012

Spore, 2008

Exercise: Generate 2D Landscape



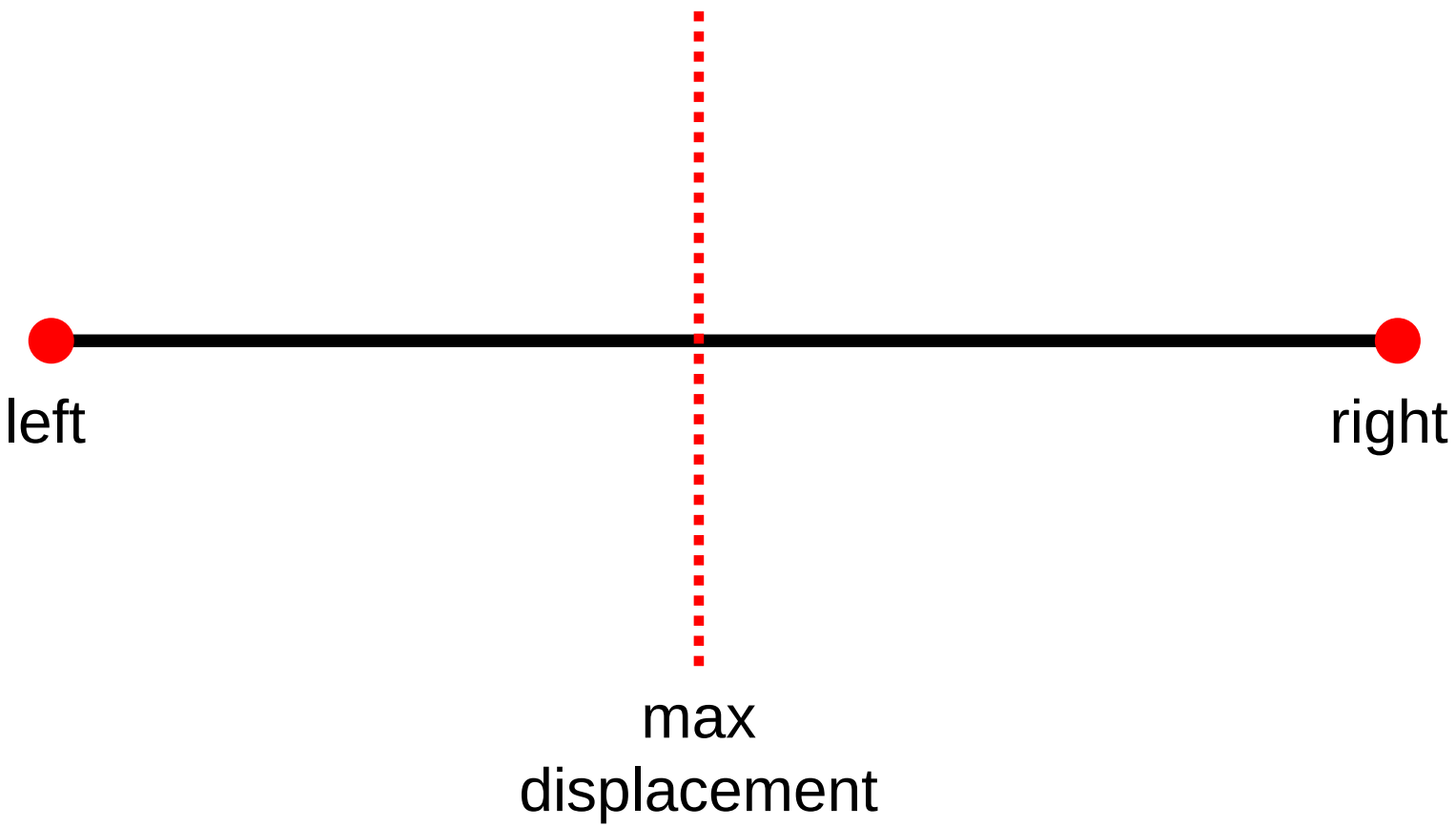
Scorched Earth, 1991
Worms, 1995

Midpoint Displacement (2D)

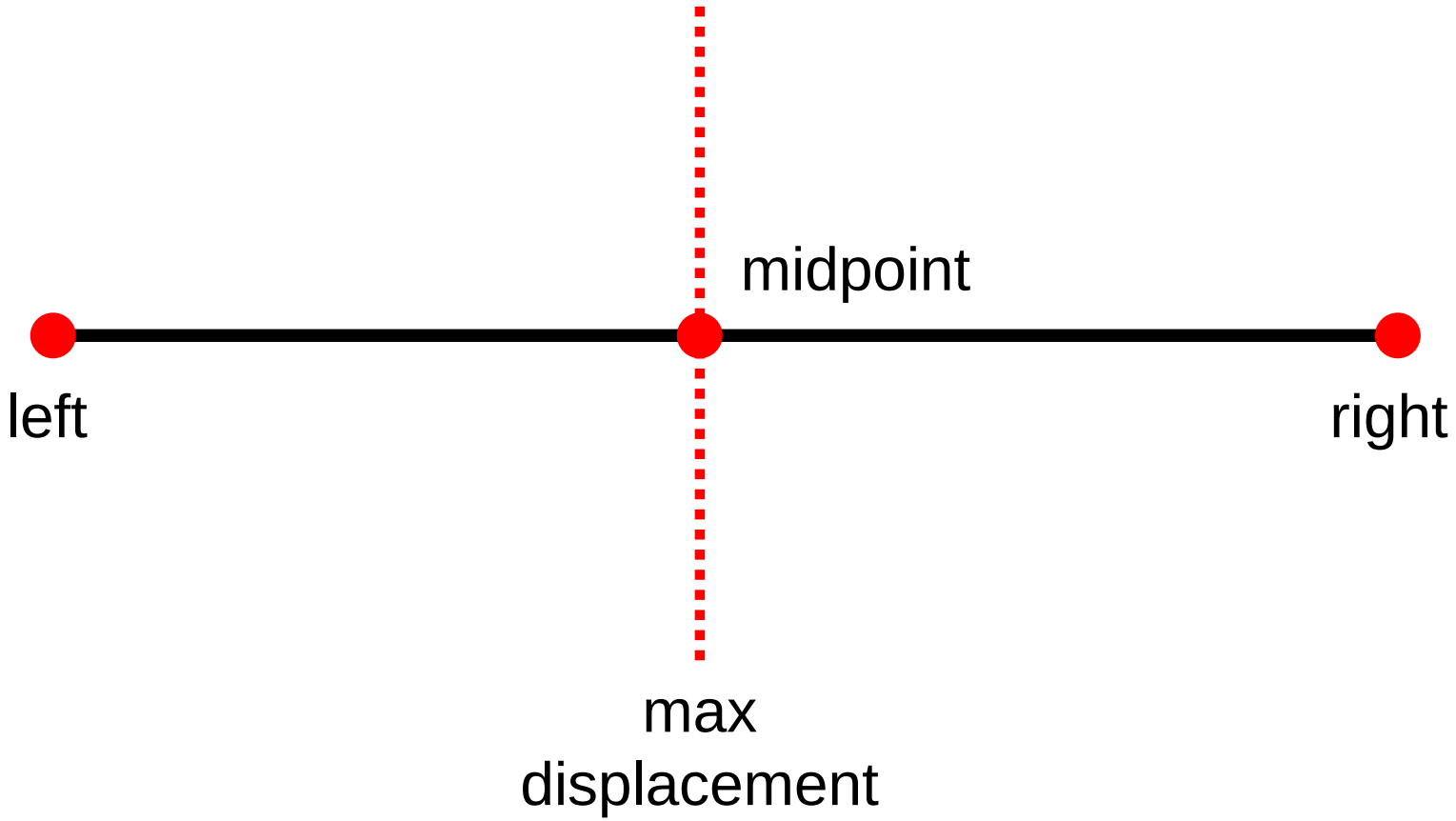
Based on generating self-similar random fractals (patterns where a smaller part appears the same as the whole).

1. Begin with a line segment.
2. Raise or lower the midpoint by a random amount.
3. Repeat this process for the left and right halves of the line segment, but reduce the amount by which the midpoint can be displaced.

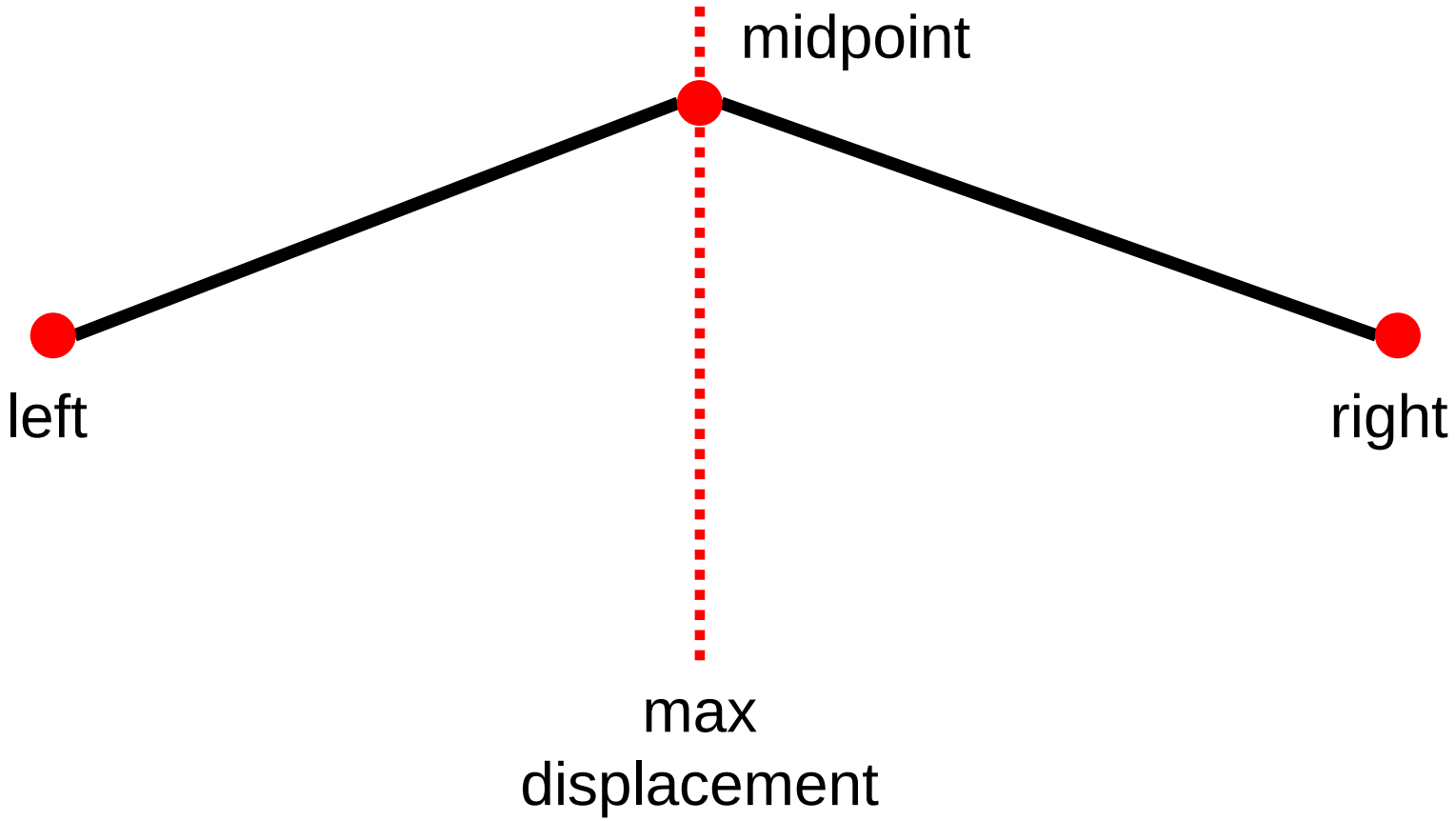
Midpoint Displacement



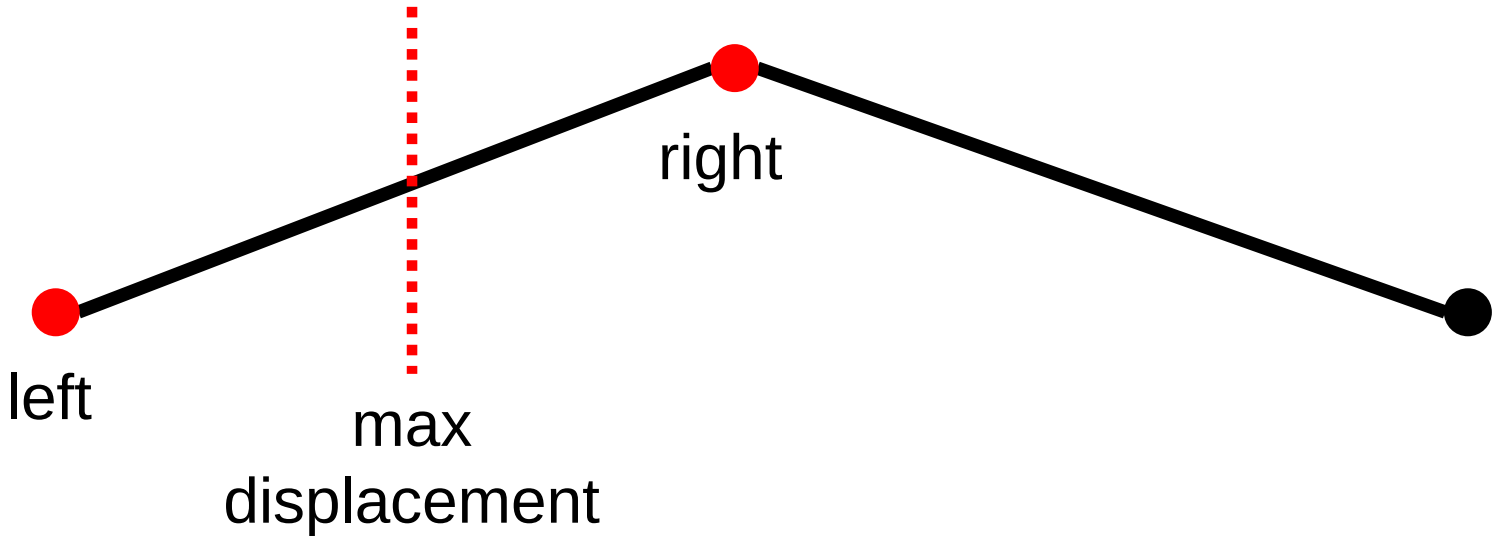
Midpoint Displacement



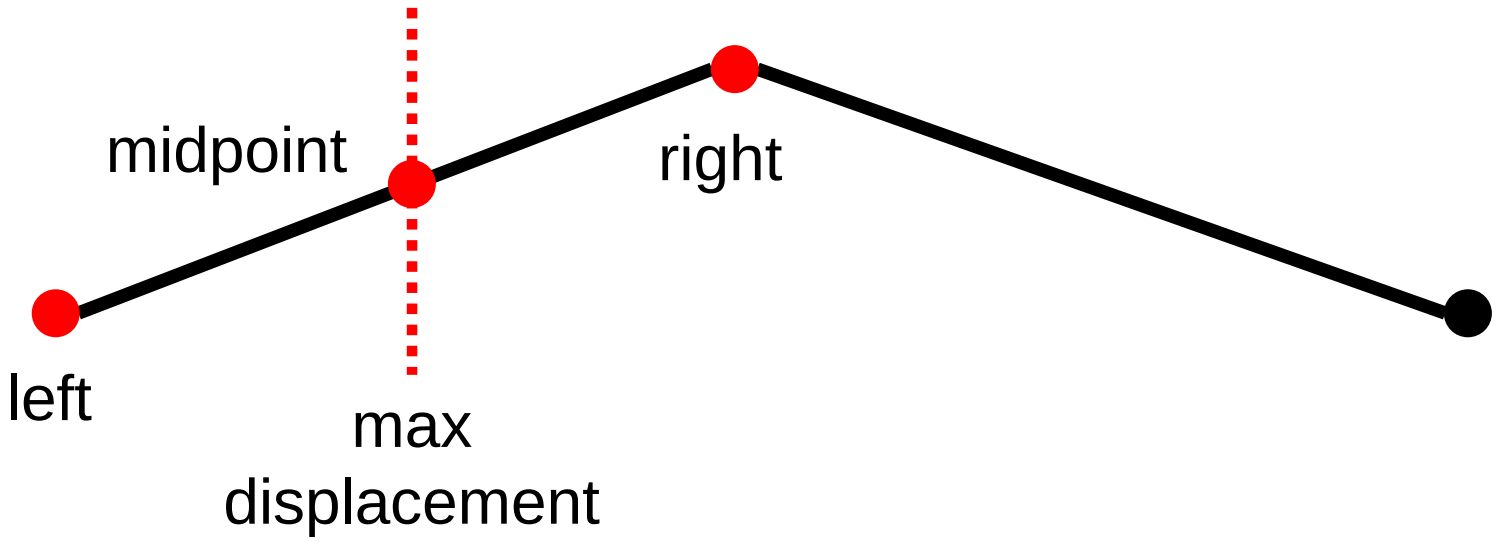
Midpoint Displacement



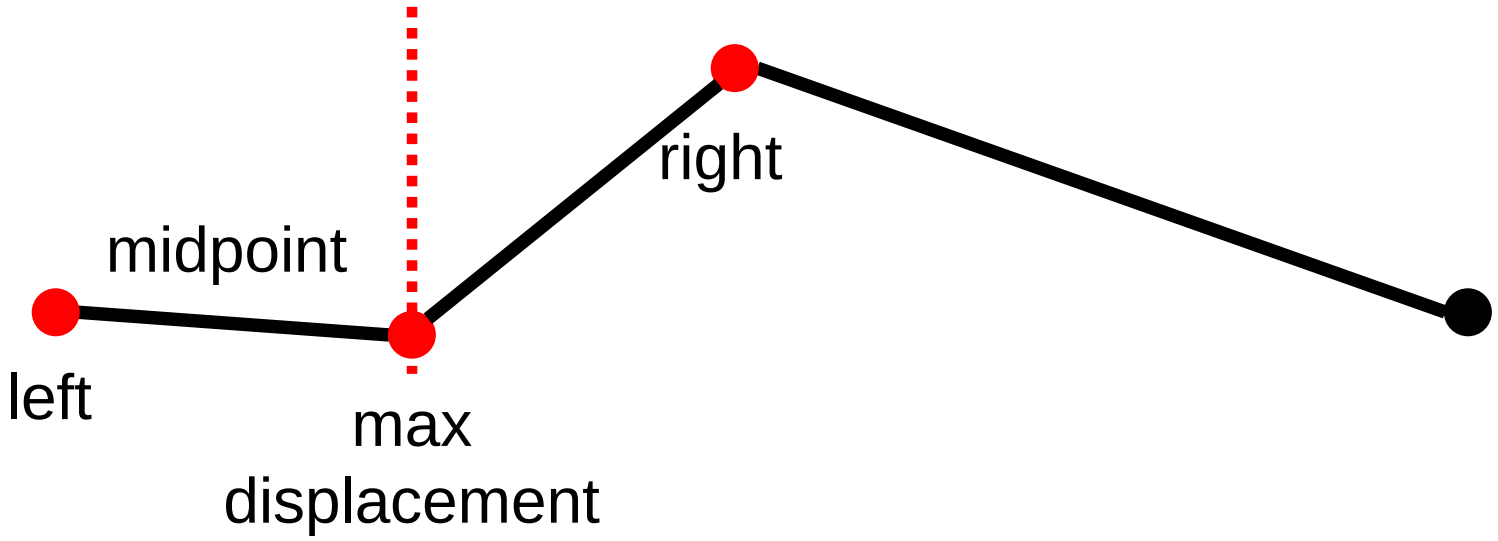
Midpoint Displacement



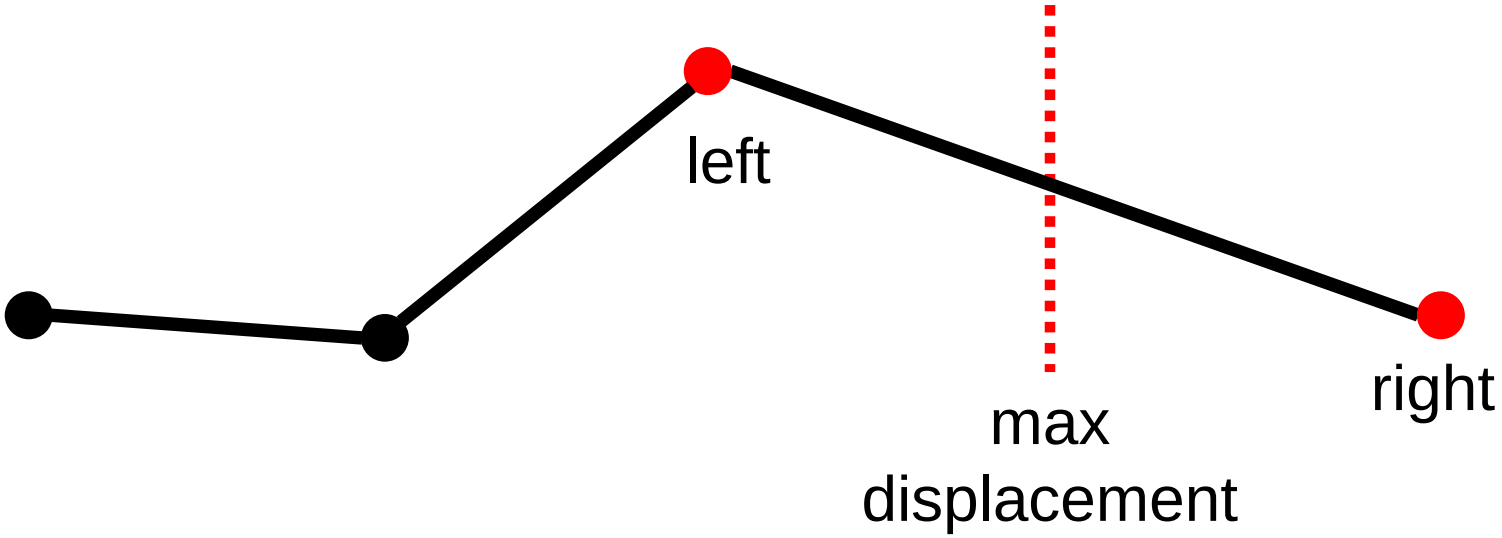
Midpoint Displacement



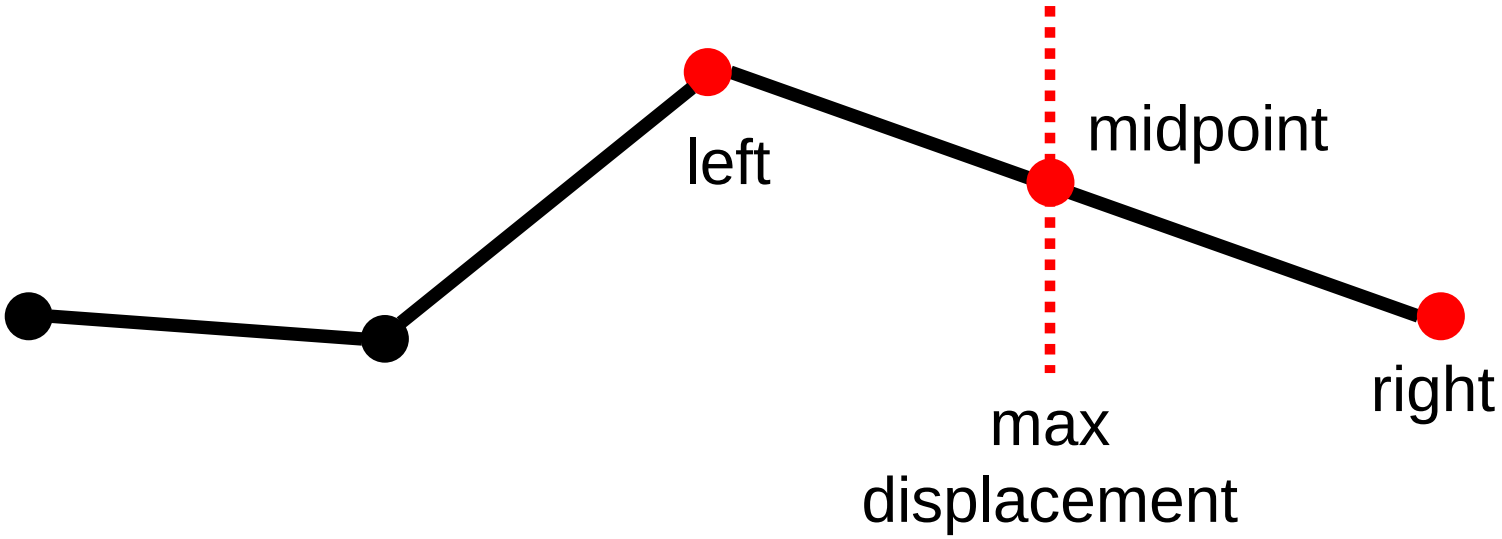
Midpoint Displacement



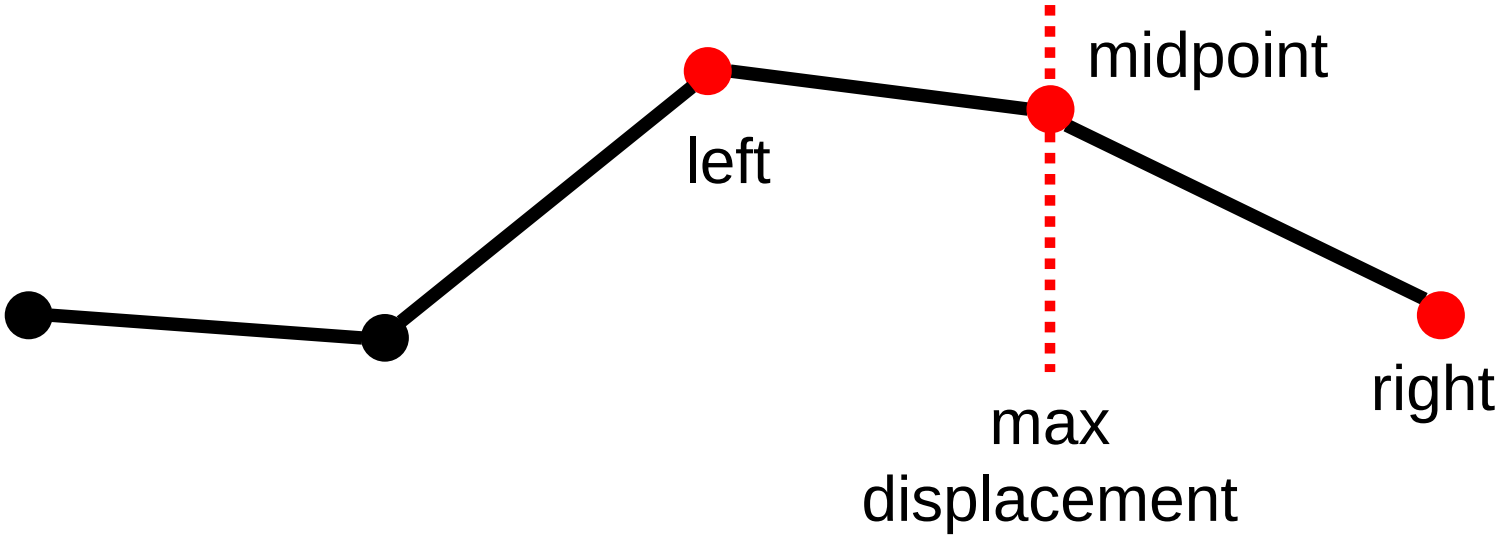
Midpoint Displacement



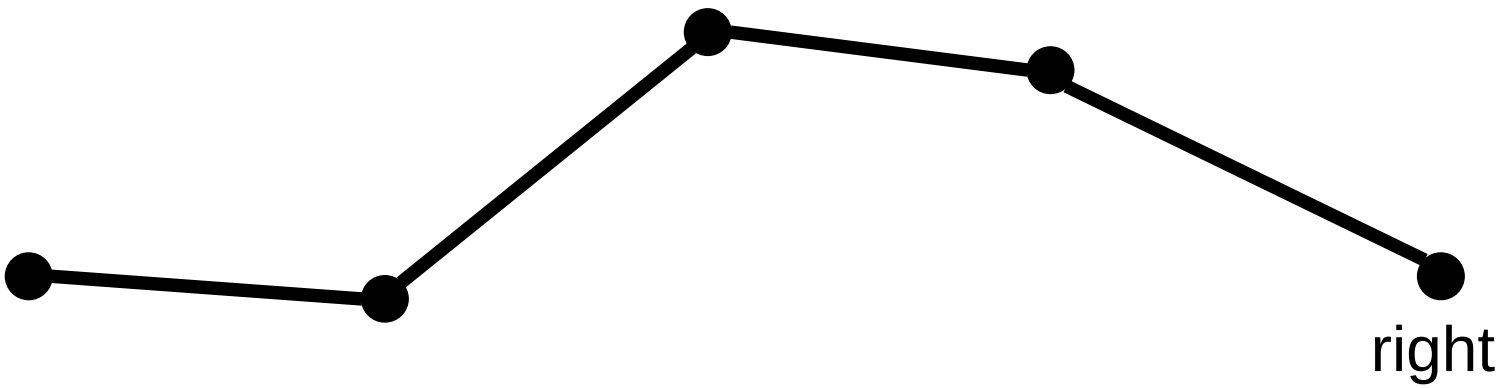
Midpoint Displacement



Midpoint Displacement

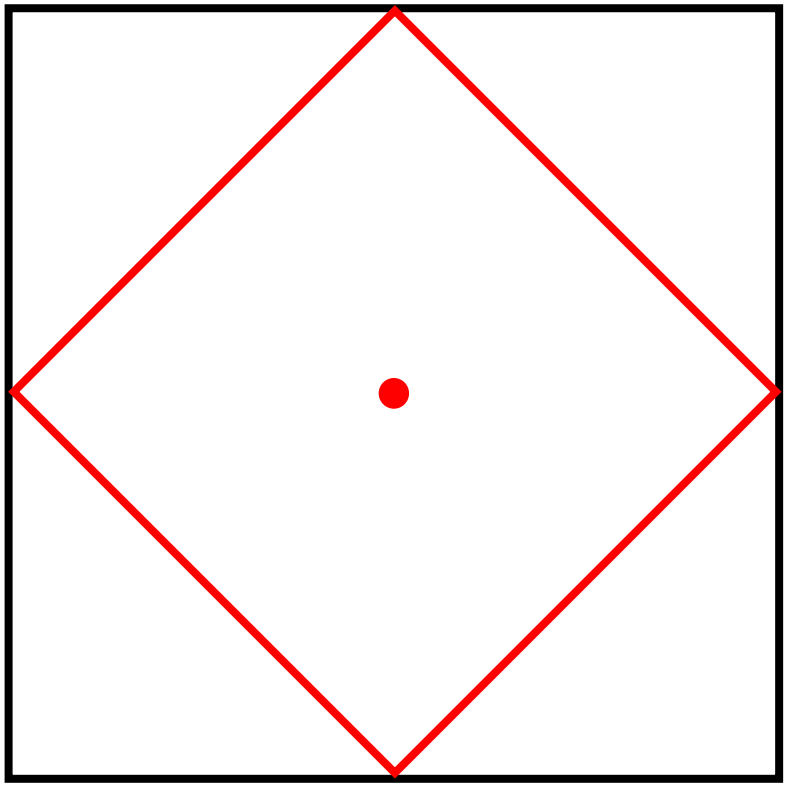


Midpoint Displacement



Midpoint Displacement (3D)

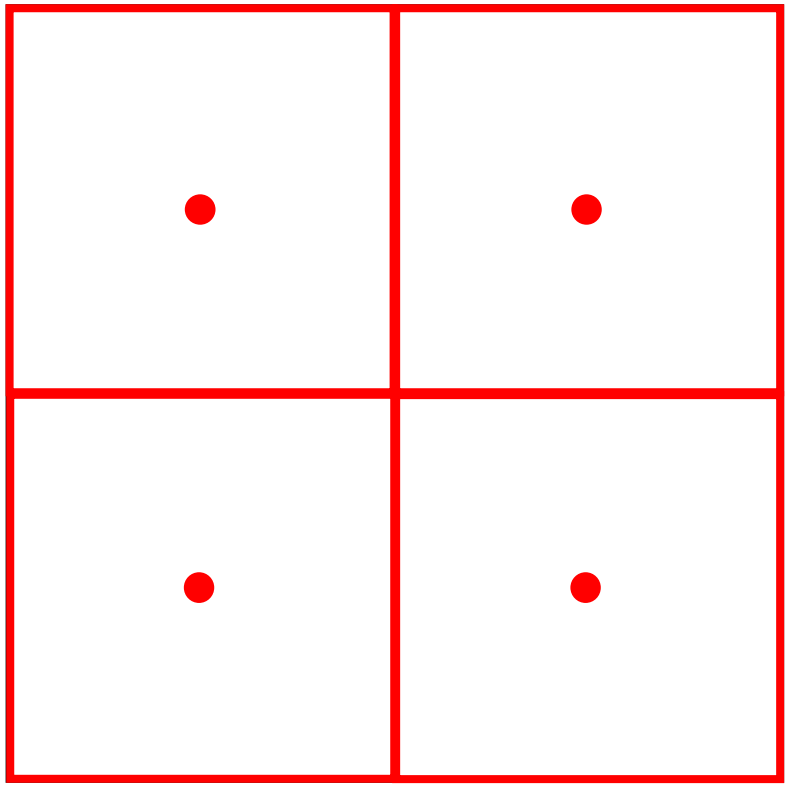
- Same basic principle, but using squares instead of lines.
- Rotate square 45° to avoid square-like artifacts (the diamond square algorithm)



diamond step

Midpoint Displacement (3D)

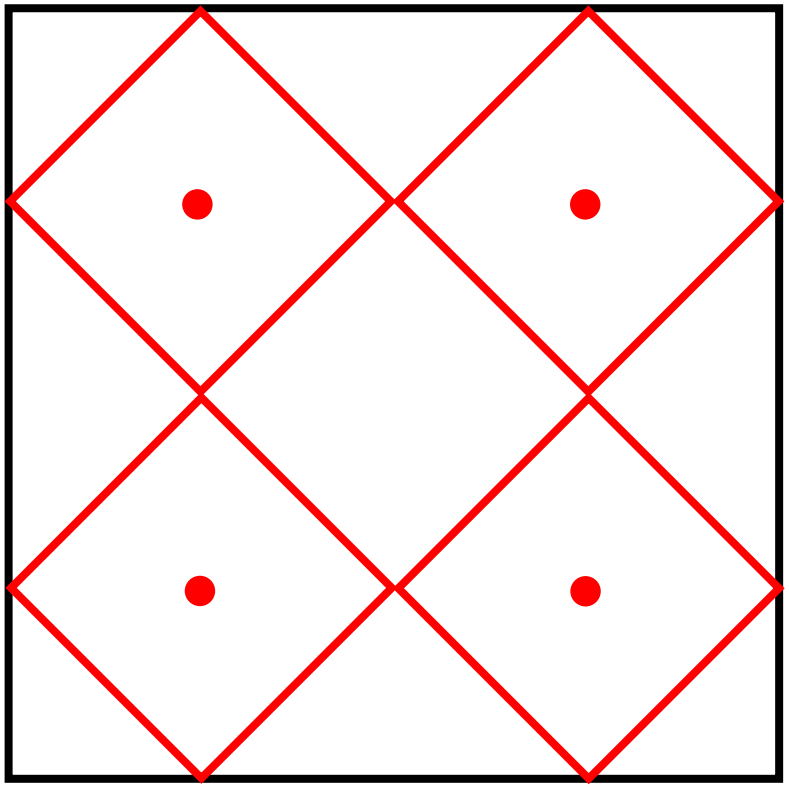
- Same basic principle, but using squares instead of lines.
- Rotate square 45° to avoid square-like artifacts (the diamond square algorithm)



square step

Midpoint Displacement (3D)

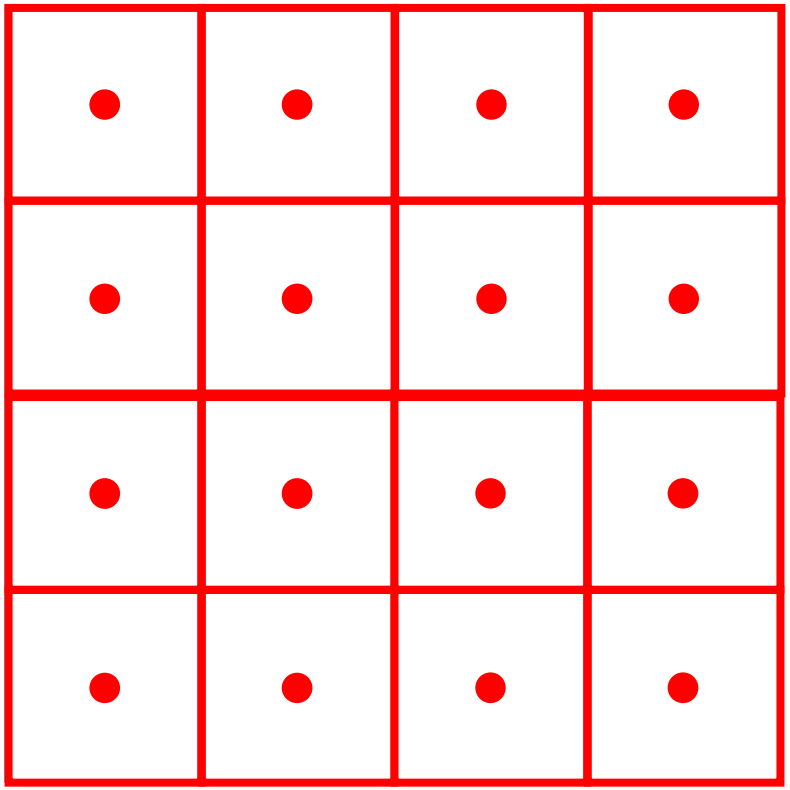
- Same basic principle, but using squares instead of lines.
- Rotate square 45° to avoid square-like artifacts (the diamond square algorithm)



diamond step

Midpoint Displacement (3D)

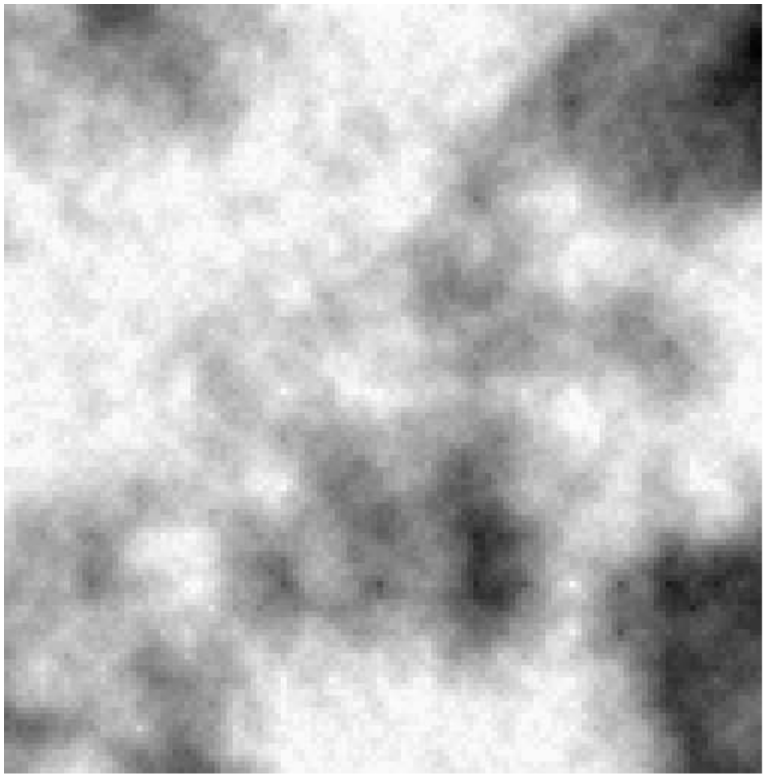
- Same basic principle, but using squares instead of lines.
- Rotate square 45° to avoid square-like artifacts (the diamond square algorithm)



square step

Midpoint Displacement (3D)

- Same basic principle, but using squares instead of lines.
- Rotate square 45° to avoid squar-like artifacts (the diamond square algorithm)



Perlin / Simplex Noise Maps

- Used in games like Minecraft
- Any position in the space can be sampled, so we don’t need to generate the whole space before the game starts (terrain can be generated on the fly).

Procedural Content Generation

Procedural content generation (PCG) is the programmatic generation of game content using a random or pseudo-random process that results in an unpredictable range of possible game play spaces.

Source: PCG Wiki (pcg.wikidot.com)

Procedural Content Generation

Procedural content generation (PCG) is the programmatic generation of game content using a random or pseudo-random process that results in an unpredictable range of possible game play spaces.

Source: PCG Wiki (pcg.wikidot.com)

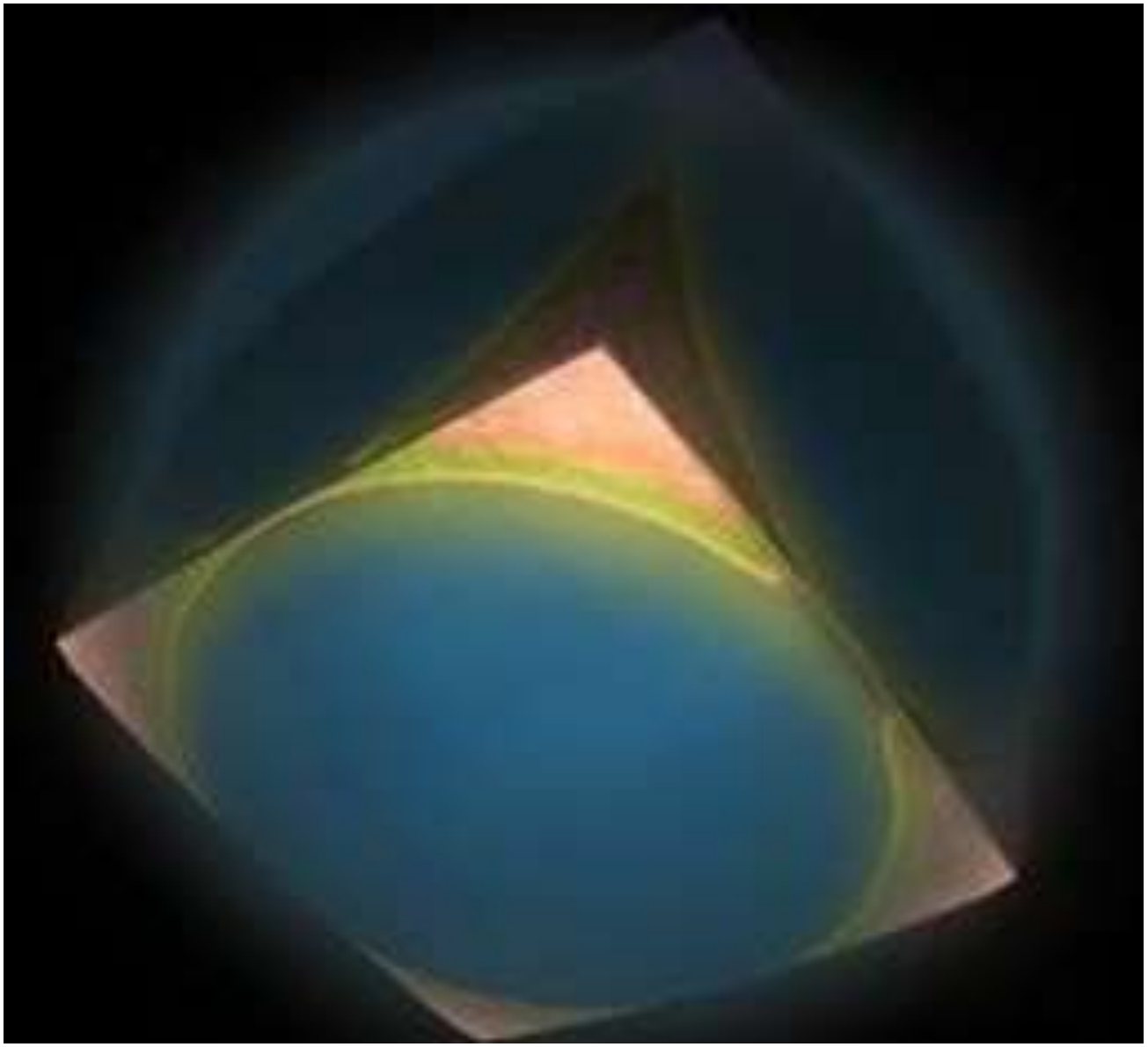
Procedural Content Generation

Procedural content generation (PCG) is the programmatic generation of game content using a random or pseudo-random process that results in an unpredictable range of possible game play spaces.

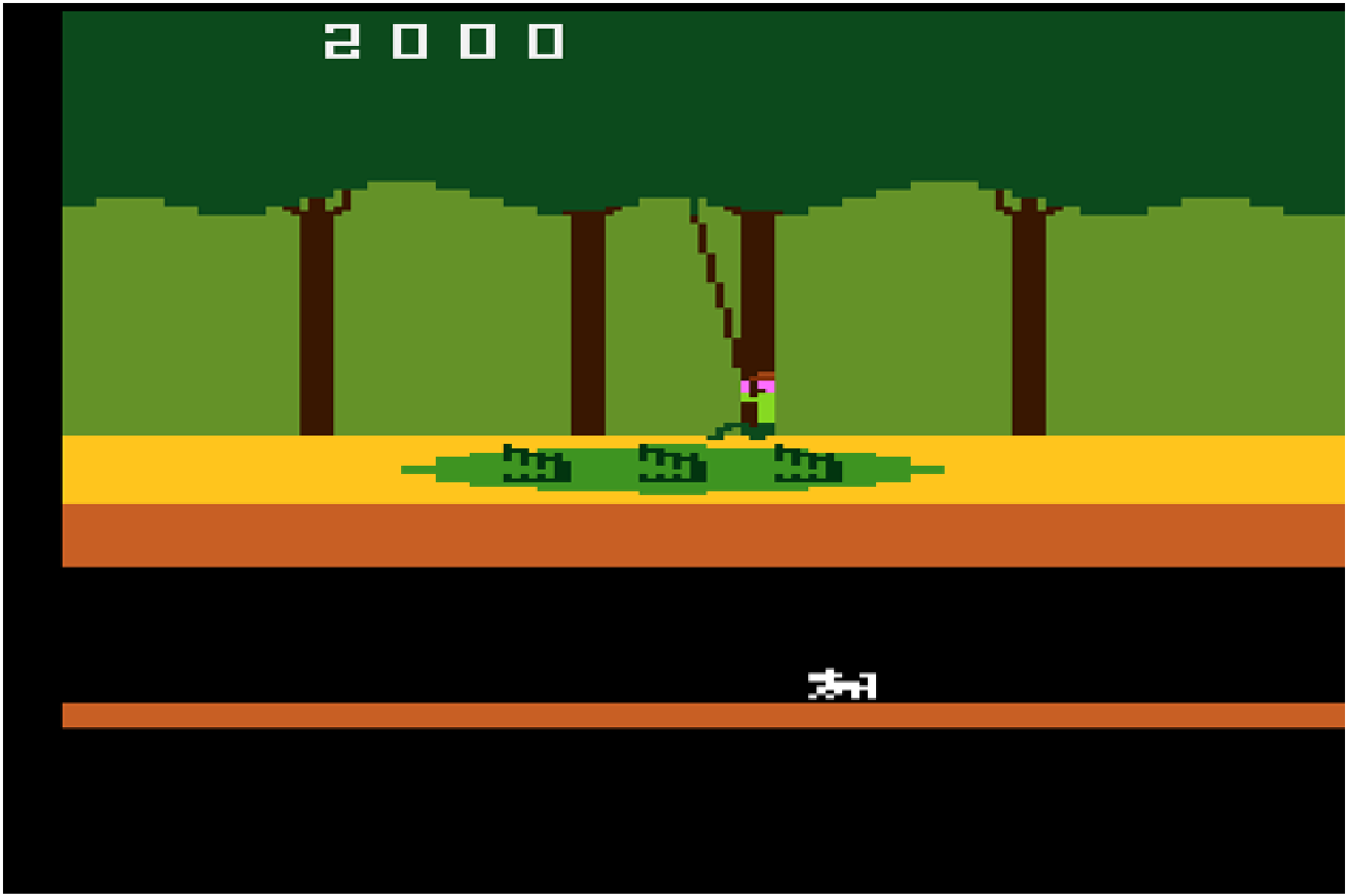
Source: PCG Wiki (pcg.wikidot.com)

Farranti Mark I Computer, 1951

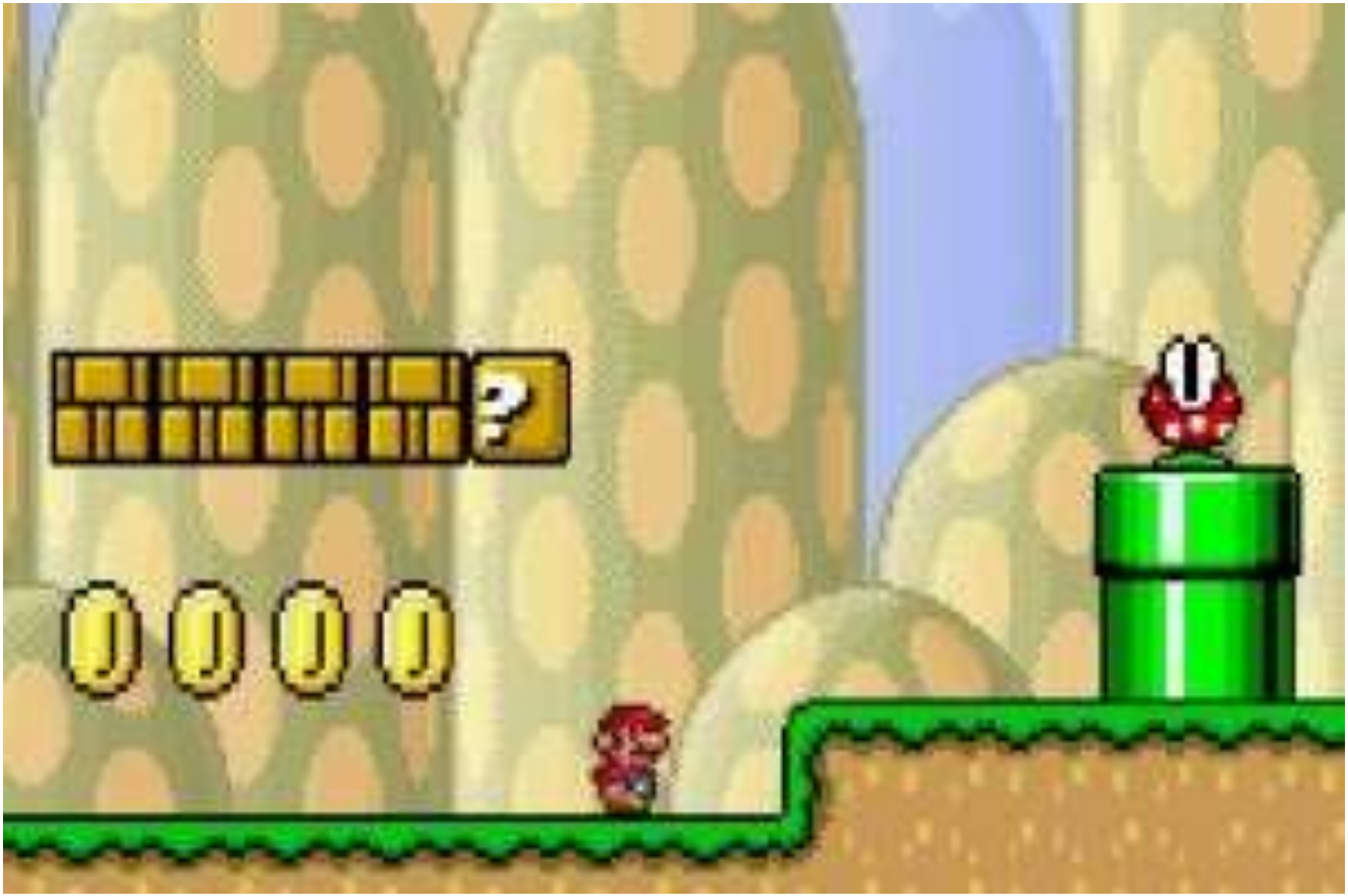
Cube planet gone wrong in *Spore*



Pitfall! 1982



Infinite Mario, 2008



Left 4 Dead 2, 2009

Procedural Content Generation

- Randomness is easy!
- Unpredictable is bad!

Procedural Content Generation

Procedural content generation (PCG) is the programmatic generation of game content guided by quality metrics.

Challenge

Designing quality metrics that constrain content generation to produce a diverse play space, all of which is meaningful.

PCG for Narrative

- Narrative Madlibs
- PCG Quests in *Skyrim*
- Beat-based model in *Façade*
- Plan-based models

The Best Laid Plans, 2014

Pros of PCG

- Reduced authoring burden
- Emergent behavior
- Increased replay value
- Thwarts walkthroughs

Cons of PCG

- Difficult to design quality metrics
- Difficulties in quality assurance
  - Varying quality of experiences
  - Difficult to provide support
- Thwarts walkthroughs

One alternative to PCG: Crowdsourcing