Procedural Content Generation in Games

based on pcgbook.com

Fachprojekt Digital Entertainment Technologies

TU Dortmund University, Germany

08/05/2017

technische universität
dortmund

Introduction

Procedural Content Generation

PCG is the algorithmic creation of game content with limited or indirect user input

content: levels, maps, game rules, textures, stories, items, quests, music, weapons, vehicles, characters, etc.

Examples

Rogue (1980)

Spelunky (2008)

Diablo (96/00/12)

Examples

Minecraft (2011)

No Man's Sky (2016)

Examples

Left 4 Dead (08/09)

Dirt 4 (2017)

Desirable Properties

- Speed

- Reliability

- Controllability

- Expressivity and diversity

- Creativity and believability

Taxonomy

- Online vs. Offline

- Necessary vs. Optional

- Degree and dimension of control

- Generic vs. Adaptive

- Stochastic vs. Deterministic

- Constructive vs. Generate-and-Test

- Automatic Generation vs. Mixed Authorship

Minecraft Settlement Generation Competition

Important Links

- Website: http://gendesignmc.engineering.nyu.edu/
- Discord: https://discordapp.com/invite/ueaxuXj
- Paper: https://arxiv.org/abs/1803.09853

Selected Objectives

- Choose type of settlement

- Use fitting material (sand in dunes, stone in mountains, snow)

- Build on ground, ideally atop hill and close to water

- Build fortification (wall / moat)

- Build food production (fields, stables)

Purely Constructive: Farming village

- Suitable house templates with modifiable
  - material
  - size

- Field templates with modifiable
  - crop
  - size

- Placement algorithm
  - Find flat surface area close to water
  - Build moat
  - Uniformly sample house positions + size
  - Place fields on flat surfaces closeby

Search-based Algorithms
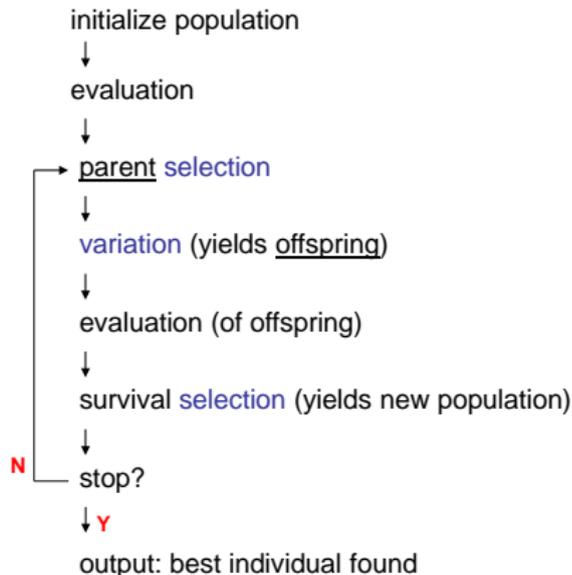
General Components

- Search Algorithm
- Content Representation
  - genotype-phenotype mapping
  - sparsity / richness
- Evaluation Functions
  - Specification
  - Direct vs. Simulation-based

StarCraft Map Generation Example

- SMS-EMOA
- vector in $\mathbb{R}^{100}$
  - start points for map features
  - all valid but many unfair
- based on positioning and paths
  - Based on expert knowledge
  - Direct (and A*)

Evolutionary Algorithms

algorithmic
skeleton

initialize population
↓
evaluation
↓
→ parent selection
↓
variation (yields offspring)
↓
evaluation (of offspring)
↓
survival selection (yields new population)
↓
N stop?
↓ **Y**
output: best individual found

technische universität
dortmund