

EVO 2009

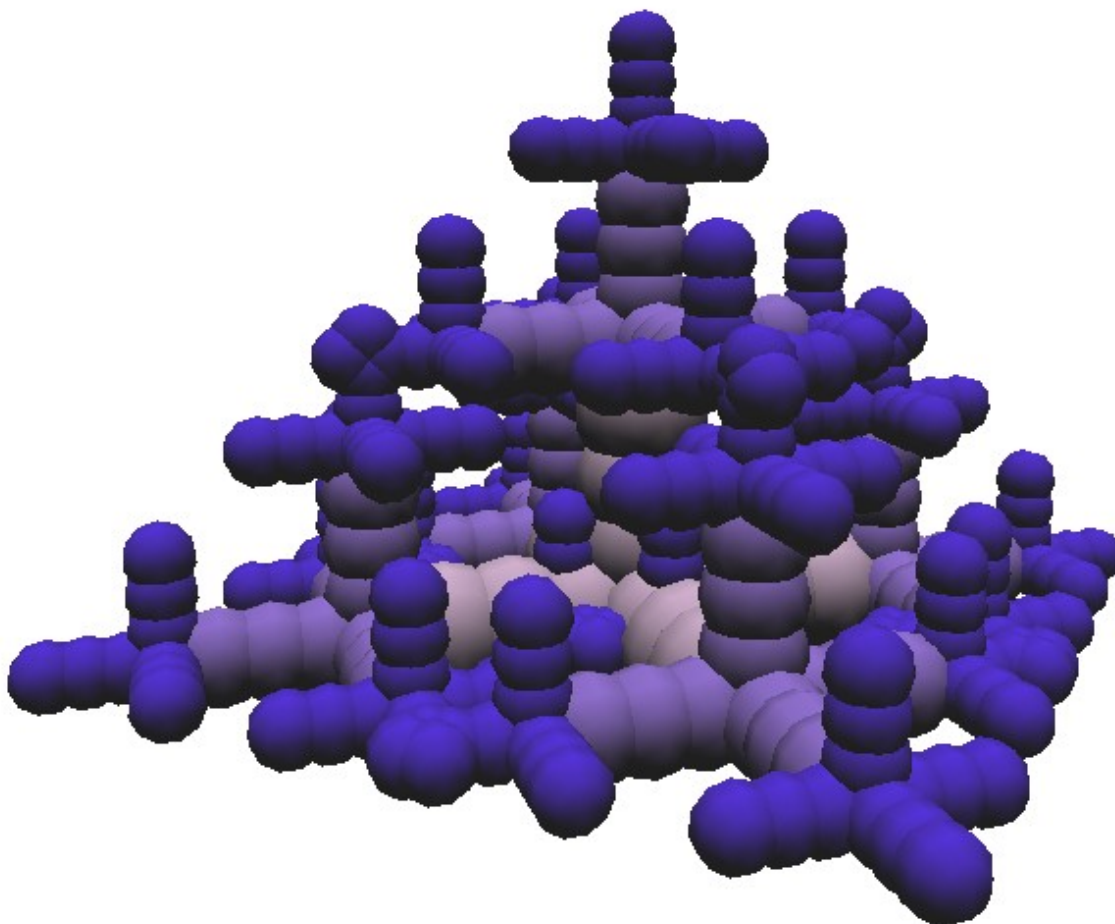
Evoluční návrh 3D objektů

Zadání

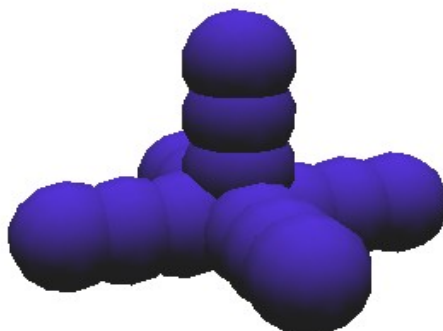
Zadáním projektu bylo vytvoření aplikace, která umožní uživateli evoluční návrh 3D objektů složených z určitých elementů (např. kostiček). Kvalita jednotlivých stupňů evoluce je pak ohodnocována přímo uživatelem.

Analýza problému

Prvním krokem bylo zvolení typu nebo třídy 3D objektů, které chceme navrhovat. Nutným předpokladem takových objektů je jejich parametrizovatelnost. Potřebujeme nějaké řídicí body, které budou daný objekt popisovat a které můžeme evolučním algoritmem optimalizovat. Já jsem proto zvolil jednoduchý fraktálovitý objekt, složený z koulí.



Tento objekt je na každé úrovni složen z následujícího elementu:



Každá úroveň se tedy větví do pěti směrů, přičemž je vždy o trochu menší, než úroveň předchozí. Zároveň jsou definovány barvy pro úroveň vnitřní a úroveň vnější. Barva úrovně mezi nimi je získána lineární interpolací.

Parametry, které objekt popisují jsou potom následující (tučně jsou zvýrazněny parametry, které se optimalizují evolučním algoritmem):

- poloha, velikost, rotace
- počet koulí v jedné úrovni
- zmenšení následujících úrovní
- **vnitřní barva, vnější barva**
- **větvení do pěti směrů**

Řešení

Z předchozího odstavce známe parametry, které chceme optimalizovat evolučním algoritmem. Složení chromozomu tedy bude následující:

Vnitřní barva	Vnější barva	Větvení do hora	Větvení doleva	Větvení doprava	Větvení do předu	Větvení dozadu
------------------	-----------------	--------------------	-------------------	--------------------	---------------------	-------------------

Vnitřní a vnější barva jsou číselné parametry typu `double` o třech složkách (červená, zelená, modrá), zatímco jednotlivá větvení jsou parametry typu `boolean`, které říkají zda se konkrétní úroveň bude větvit do patřičného směru. Toto větvení se optimalizuje pro každou úroveň zvlášť, jak bude ukázáno později.

Takto sestavený chromozom je jádrem celé optimalizace, jsou na něj aplikovány obvyklé operátory mutace a křížení.

Fitness funkce je v našem případě triviální co se kódu týče, vrací hodnotu nastavenou uživatelem pro konkrétní chromozom.

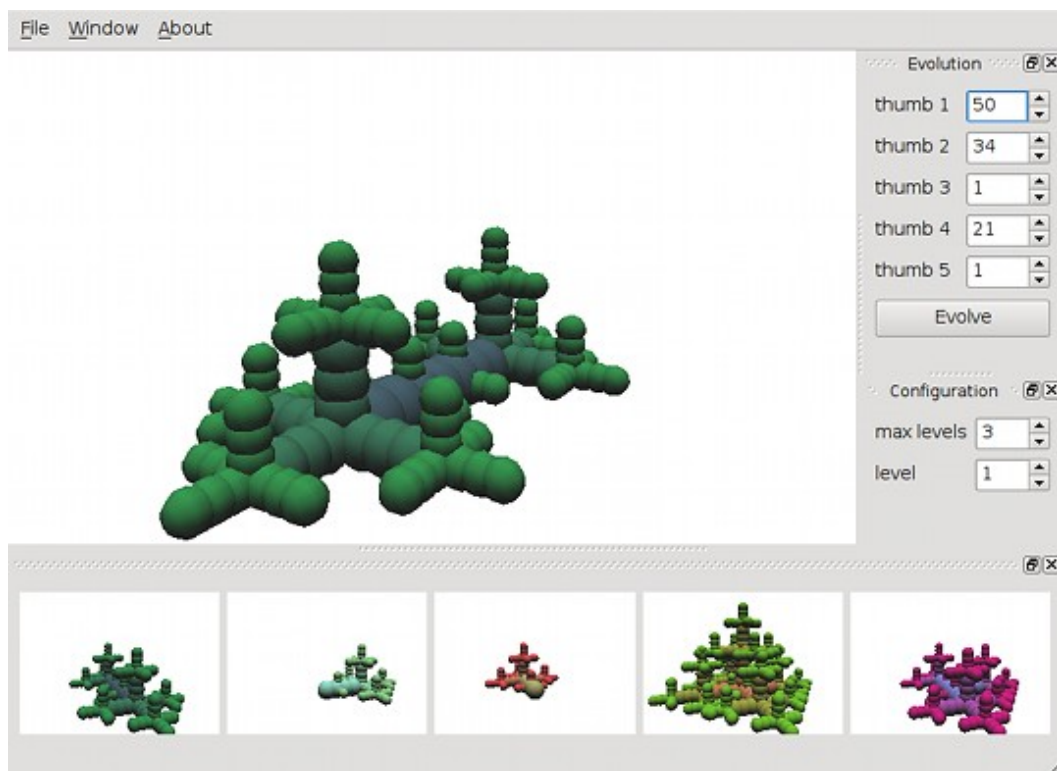
Celá optimalizace je pak prováděna na populaci o pěti členech, přičemž uživatel ji má možnost po každém kroku patřičně ohodnotit. Na tomto místě by stálo za úvahu jak velkou výpovědní hodnotu má takto malá populace v kombinaci s krokovanou evolucí, nicméně experimenty ukázaly, že řešení konverguje směrem daným uživatelem v rozumně krátké době docela pěkně.

Velikost populace a krokování bylo voleno s ohledem na pohodlí práce a ovládání pro uživatele.

Aplikace

Aplikace, která toto řešení implementuje, byla napsána v Javě a využívá knihovny Qt Jambi (grafické uživatelské rozhraní, <http://www.qtsoftware.com/downloads>), JOGL (OpenGL vykreslování, <https://jogl.dev.java.net/>) a JGAP (evoluční algoritmy, <http://jgap.sourceforge.net/>).

Její podoba je následující:



V prostřední části je vidět zvětšený náhled jednoho řešení z populace, ve spodní části jsou potom náhledy celé populace. Objekty v těchto oknech je možné rotovat stisknutím levého tlačítka myši a jejím tažením v horizontálním směru nebo měnit jejich velikost stisknutím pravého tlačítka a tažením ve vertikálním směru. Stisknutím prostředního tlačítka na některém náhledu se tento náhled zobrazí zvětšený v prostřední části aplikace.

V pravé části aplikace máme možnost ovládat evoluci a nastavení objektů. U jednotlivých náhledů z populace je možné nastavit jejich fitness hodnotu, kde 1 je nejhorší. Stisknutím tlačítka *Evolve* se provede jeden krok evoluce.

V pravé spodní části je možné nastavit maximální počet větvení (projeví se v dalším kroku evoluce) a právě optimalizovanou úroveň větvení. Tato hodnota má význam pro optimalizaci větvení do různých stran, kdy vyvíjí jednotlivé úrovně odděleně.

Závěr

Výsledkem projektu je aplikace, která umožňuje uživateli navrhovat 3D objekty s využitím evolučních algoritmů. Celý proces optimalizace je ovládán cítěním uživatele a tudíž by měl vést k objektům, které si přeje.

V oblasti výběru vhodných objektů k evoluci a jejich parametrickém popisu je velký prostor pro experimentování. V mém případě by mohlo být zajímavé např. optimalizovat barvu a tvar objektu zvlášť, nebo najít nějakou funkci pro popis větvení skrze všechny úrovně, aby se nemusely jednotlivé úrovně vyvíjet samostatně. Naopak experimenty s optimalizací velikostí jednotlivých úrovní se ukázaly jako nepříliš vhodné, protože ztěžovaly orientaci v objektu.

Nakonec by stálo za promyšlení efektivnější uživatelské rozhraní aplikace, kdy by uživatel nemusel psát fitness hodnoty pro jednotlivé řešení, ale mohl je například tažením řadit dle priority.

Nakonec ještě několik příkladů výstupu z aplikace.

