
Hilbert space-filling curve

Datum: 13.12.2014
Autor: Bc. Jan Urubek
Login: URU0006

Space-filling curve

Křivka vyplňující prostor je křivka, která beze zbytku vyplňuje oblast n -rozměrného prostoru. Jednoduše řečeno 1D křivka vyplní n -rozměrný prostor. Jedná se o fraktál, protože dimenze výsledného útvaru je (ostře) větší než dimenze křivky.

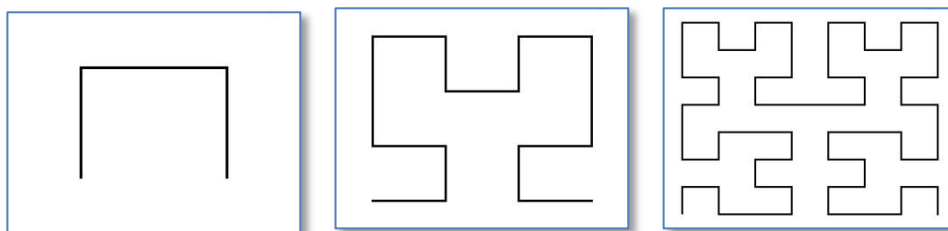
Vlastnosti prostor vyplňujících křivek:

- patří mezi fraktály
- jsou si sobě podobné
- jsou invariantní vůči velikosti
- jsou nekonečně dlouhé
- přestože v zobrazení prvních iterací mnohých z nich převládají úsečky, limitně úsečky neobsahují
- přestože technicky vzato se musí v každém bodě protínat, podle většiny zadání se neprotíná vůbec
- dimenze výsledného útvaru je (ostře) větší než dimenze křivky.

Hilbert space-filling curve

Hilbertova křivka je fraktálová, plochu-vyplňující křivka, jejíž dvourozměrnou variantu jako první popsal německý matematik David Hilbert v roce 1891.

Hilbertova křivka vytvořená v n -té iteraci ($n > 1$) se získá ze čtyř Hilbertových křivek vytvořených v $n-1$ iteraci, přičemž tyto křivky jsou odpovídajícím způsobem orotovány a zmenšeny na polovinu (tím se dodrží velikost původního obrázku).

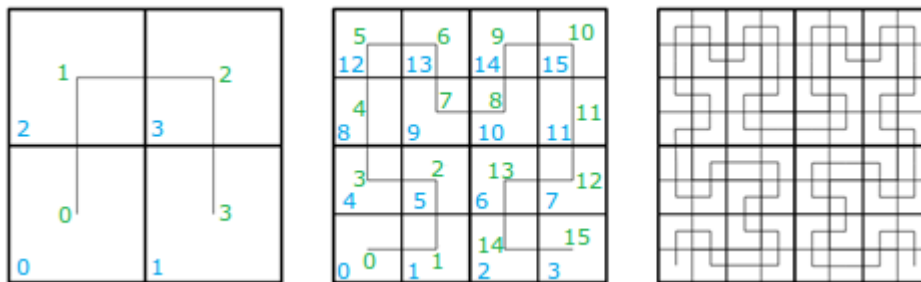


První tři iterace Hilbertovy křivky

Realizace v aplikaci

Pro generování Hilbertovy křivky byl zvolen jazyk C++ a pro následnou vizualizaci knihovna Qt obsahující widget Graphics View, který umožňuje vykreslování jednoduchých čar.

Samotným základem algoritmu je rozrašťovaná plocha na 4^N políček, kde N reprezentuje počet iterací křivky.



Rozrašťovaná plocha pro vykreslení iterací Hilbertovy křivky.

Jak lze vidět z obrázku výše, veškeré vrcholy dané křivky se nacházejí ve středu políček. Programově bylo nutné vytvořit novou datovou strukturu `workPoint`, obsahující :

- **Vector2 center;** - dvoudimenzionální pozice bodu (vrcholu)
- **int drawnIndex;** - index bodu na plátně (znázorněno modrou barvou)
- **int curveIndex;** - index určující pořadí vykreslovaného bodu (znázorněno zelenou barvou)

Samotné generování křivky bylo založeno na dynamických datových polích typu `std::vector` obsahujících prvky výše zmíněné struktury `workPoint` a implementací transformačních metod nad nimi (rotace o 90° , zrcadlení, atd..).

Základem bylo nadefinování vzhledu křivky při iteraci 1. Další následující iterace byly realizovány tak, že se vzal výsledný tvar (vektor) z minulé iterace a čtyřikrát se zkopíroval a vhodně narotoval do již rozrašťované plochy. Rovněž bylo nutné každému novému zkopírovanému tvaru zvýšit vykreslovací indexy o velikost vektoru z minulé iterace.

Výstupem tohoto algoritmu byl vektor o velikosti 4^N , kde N reprezentuje počet iterací. Tento vektor bylo nyní nutno seřadit vzestupně podle vykreslovacích indexů (`curveIndex`). Jakmile byl seřazen, stačilo jej procházet a vykreslovat čáry z předcházejícího bodu do aktuálně iterovaného.