

## Posudek oponenta na diplomovou práci

**Diplomantka:** Bc. Tatyana Peshkova

**Téma:** Využití Voronoi diagramu koulí pro modelování granulí

**Vedoucí:** Prof. Dr. Ing. Ivana Kolingerová

DP testuje potenciál využití pokročilé geometrické struktury VDS (Voronoi diagram koulí) pro dvě geometrické úlohy:

1. Plnění prostoru uvnitř kontejneru (krychle, válec nebo obecný objekt) neprotínajícími se koulemi.
2. Simulace pohybu granulí různých velikostí.

Zatímco využití VDS pro plnění koulemi se ukázalo jako velmi slibné, využití pro simulace pohybu granulí se ukázalo jako nevhodné.

**Struktura:** DP shrnuje stav poznání o vlastnostech VDS a algoritmech konstrukce. Popisuje známou aplikaci plnění válce koulemi stejné velikosti a analýzu volného prostoru pomocí VDS. Je zmíněn i konkurenční plnicí algoritmus Proto-Sphere. Následuje popis výpočetních knihoven třetích stran (konstrukce VDS a simulace kapalin pomocí částic stejné velikosti). Kapitola 6 už je vlastní řešení problému – obsahuje algoritmus plnění prostoru a algoritmus pro simulaci pohybu částic různých velikostí. Následují experimenty a závěr.

**Plnění geometrických objektů koulemi:** Jsou prezentovány dva algoritmy – jeden pro stejně velké koule a jeden pro koule s poloměrem z daného intervalu. Podstata je ale stejná – udržování VDS pro koule tvořící výplň a vkládání nových vyplňujících koulí tak, aby jejich střed byl ve vrcholu (případně na hraně) VDS. Lze pochopit, že po každém vložení je potřeba přepočítat VDS, protože zvolená knihovna (ani žádná jiná) nepodporuje vkládání jednotlivých koulí, a nejde nám ani tak o čas jako spíše o kvalitu. Už méně pochopitelná je práce s množinou kandidátů na vložení nových koulí – pokud tedy bylo cílem vkládat koule od největší možné (zmíněno na str. 30). Algoritmus (6.1.2 na str. 32) totiž provádí cyklus, kde vytvoří množinu kandidátů (vrcholy VDS), tu pak celou zpracuje (v pořadí od max. k min. poloměru), vytvoří další množinu kandidátů, zase ji zpracuje, atd. Tím se do výplně mohou postupně dostávat jak velké, tak velmi malé koule. Správně by se měla pro výběr kandidáta na vložení do výplně použít prioritní fronta. Podmínka na zastavení algoritmu „dokud lze plnit“ je slabá (pokud min. povolený poloměr plnicí sféry bude skoro nula, pravděpodobně dojde paměť). V experimentech se uvádí čas výpočtu (o ten ale až tak nejde), ale nikde tam není míra kvality zaplnění (např. podíl objemu prostoru zaplněného koulemi k celkovému objemu kontejneru).

**Simulace pohybu částic (granulí) různých velikostí:** Pohyb středů částic (bodů) je řízen externí knihovnou pro SPH (Smooth Particles Hydrodynamics) a upravován algoritmy z DP tak, aby se koule částice odrazila od koulí ostatních částic, případně od stěny kontejneru. VDS se používá pro získání množiny kandidátů na kolizi pro každou částici. Tolik záměr, teď problémy:

1. Metoda SPH pro částice definuje poloměr jejich vlivu a počítá s tím, že se částice mohou dostat blízko sebe. Konvolucí se pak určí nové vlastnosti částice. Pokud se budeme snažit zabránit kolizím, výrazně tím zasáhneme do simulace SPH (omezíme ovlivňování částic mezi sebou).
2. V algoritmech (6.2.1-3) je potřeba v daném časovém okamžiku aktualizovat pozice všech částic a vyřešit jejich kolize. Úzkostlivé vyhýbání se kolizím stylem „jedna pohybující se částice vs. nepohybující se ostatní“ (str. 39 – obr. 6.11) není moc přesvědčivé, ostatní částice se přeci také pohybují.
3. Výpočet kompletního VDS při posunutí každé jedné částice rozhodně není vhodné. Lepší by bylo spočítat VDS pro všechny částice v daný časový okamžik a pak tento VDS použít jako datovou strukturu pro následující (dostatečně krátký) časový úsek.

4. Chyba při výpočtu VDS by neměla zastavit částici. Takovou částici bych doporučil explicitně považovat za kandidáta na kolizi s jakoukoliv jinou částicí.
5. Jak bylo patrné z předběžných měření (str. 23 - Tab. 4.1), existující metoda pro získání sousedů buňky VDS nebyla vůbec vhodná. I přesto však byla později použita.
6. Alg. 6.2.1-3 bych spojil do jednoho (liší se pouze typem použitého kontejneru).
7. Alg. 6.2.3 – řešení kolize částice s trojúhelníkovou sítí – není řečeno, jak se v síti najde kolizní trojúhelník.

#### Drobné výhrady:

Kapitoly 4 a 5 (výpočetní knihovny) – Uvedení tolika zdrojových kódů v DP je dle mého názoru zbytečné a ve zbytku práce se skoro nevyužívá. Rovněž zbytečné je i tvrzení na straně 20, že knihovna je implementována jako black-box a že pro její použití je na ni třeba nastavit reference. Příložený SW – přes prvotní problémy (chybějící závislosti, desetinná tečka/čárka) se jej podařilo spustit a něco spočítat. Práce v něm je poněkud obtížná, diplomantka musela mít neuvěřitelnou trpělivost a odhodlání. Chybná citace na str. 13 (Místo [7] má být [1]) a na str. 25. Pochybuji o tom, že experimenty byly prováděny na CPU o frekvenci 240 GHz (Abstract, str. 22, 43). Je zbytečné v tabulkách uvádět přesnost času na 3 desetinná místa, když je hodnota v řádech tisíců (např. str. 47, Tab. 7.3). V řešení kolize pohybující se koule vs. trojúhelník (str. 39 - obr. 6.12) je malá chyba – paprsek nemusí protnout trojúhelník, aby nastala kolize. Vzorec 6.1 na str. 29 pro reprezentaci hran VDS je špatně. Ve skutečnosti se má jednat o racionální kvadratickou Bézierovu křivku, která jde převést na obyčejnou kubickou (pro snadnou vizualizaci).

#### Otázky k obhajobě:

1. Předpokládejme, že chceme plnit krychli stejně velkými koulemi tak, aby se žádné dvě neprotínaly (ale mohou se dotýkat) a aby žádná z nich neprotínala žádnou stěnu krychle (ale může se dotýkat). Známe délku hrany krychle (např. 60) a poloměr koule (např. 3). Kvalitu zaplnění budeme hodnotit jako poměr objemu prostoru zabraného všemi koulemi k objemu prostoru zabraného celou krychlí (0 nejhorší, 1 nejlepší). Jak obстоjí Vaše strategie plnění (výsledky z Tab. 7.3 na str. 47) oproti jednoduché strategii plnění v pravidelné mřížce (jako kdybychom je skládali do trojrozměrné šachovnice)? Dokážete uvést lepší odhad (číslo menší než 1) na maximální dosažitelnou kvalitu?
2. Ve Vašich algoritmech vybíráte kandidáty na plnicí koule nejen ve vrcholech VD, ale i na hranách. Jakým způsobem určujete střed kandidáta – je vybrán jako náhodný bod na hraně, nebo podle nějakého kritéria?

Práci **doporučuji k obhajobě** a navrhuji hodnocení známkou „dobře“ (C).

V Plzni dne 29. 8. 2012

Mgr. Martin Maňák  
Katedra informatiky a výpočetní techniky  
Západočeská univerzita