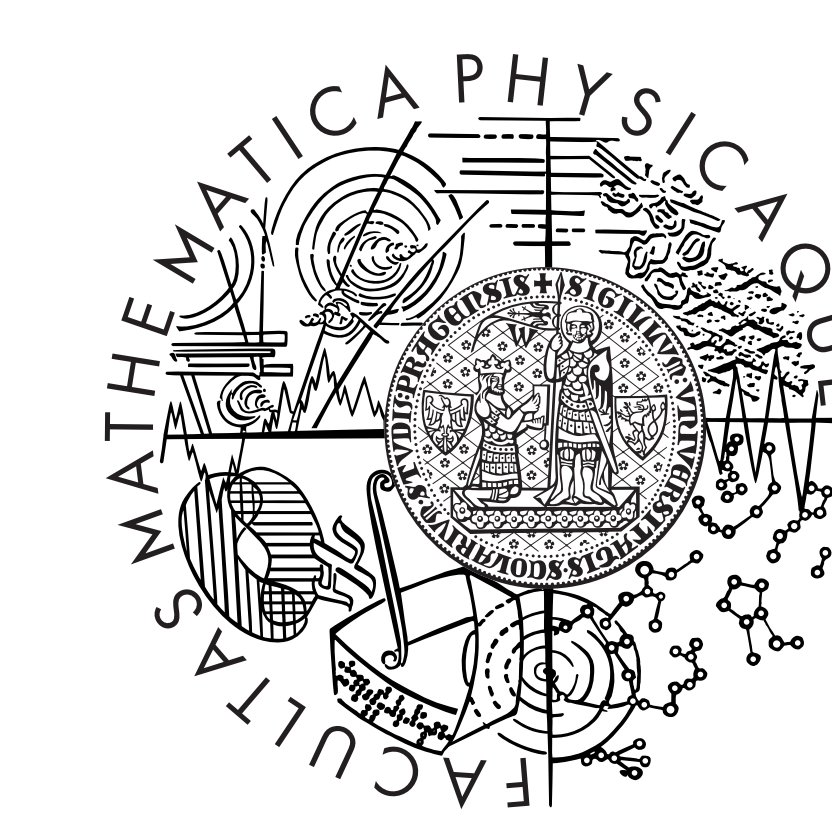


Podpora fraktúr pre jbox2d engine

Marek Beňovič

Katedra distribuovaných a spoľahlivých systémů, Matematicko-fyzikální fakulta, Univerzita Karlova v Praze

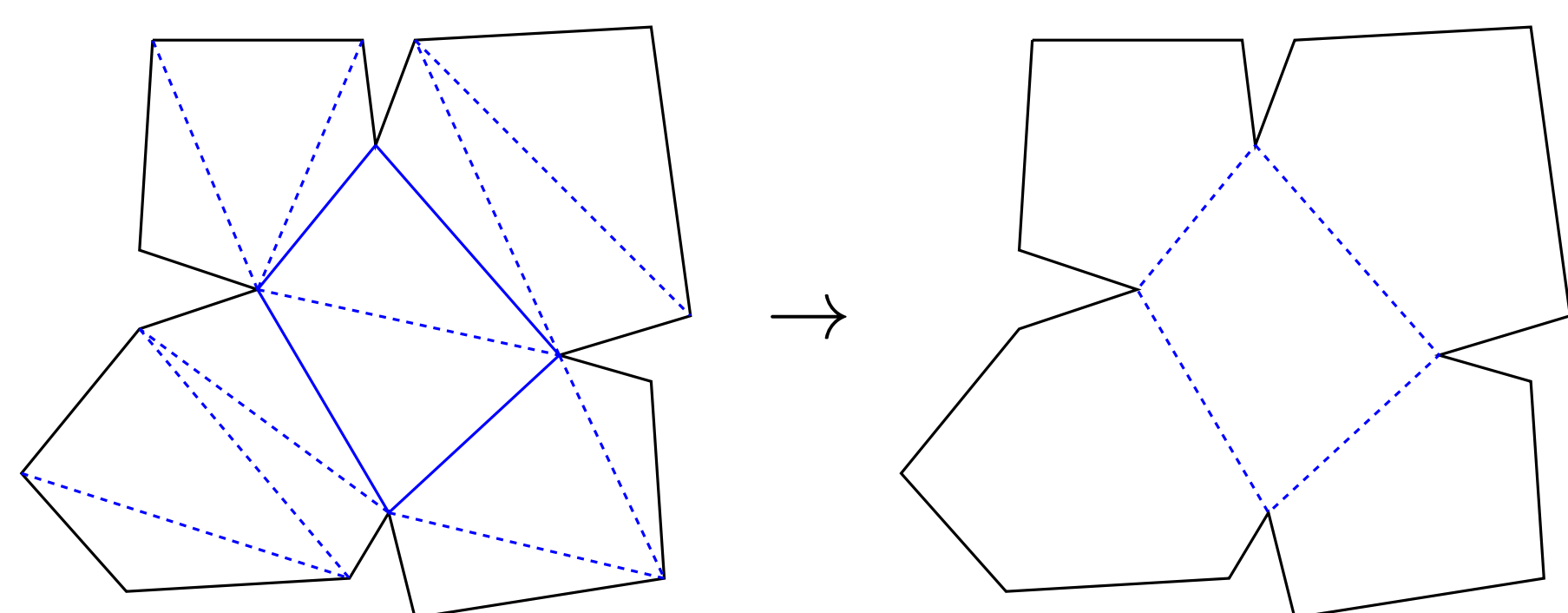


Ciele práce

JBox2D [1] je herný engine simulujúci fyziku pevných telies a kvapalín v 2D priestore implementovaný v jazyku *Java*. Práca poskytuje rozšírenie enginu JBox2D umožňujúce triedenie telies po ich vzájomnej kolízii. Dôraz je kladený na nízke nároky na výkon procesora zaručujúce plynulosť činnosti algoritmu v reálnom čase a prirodzenosť priebehu procesov triedenia. Algoritmus taktiež poskytuje možnosť definovať materiály telies a nastavovať ich vlastnosti. Je k dispozícii jednoduché programátorské rozhranie založené na logike knižnice.

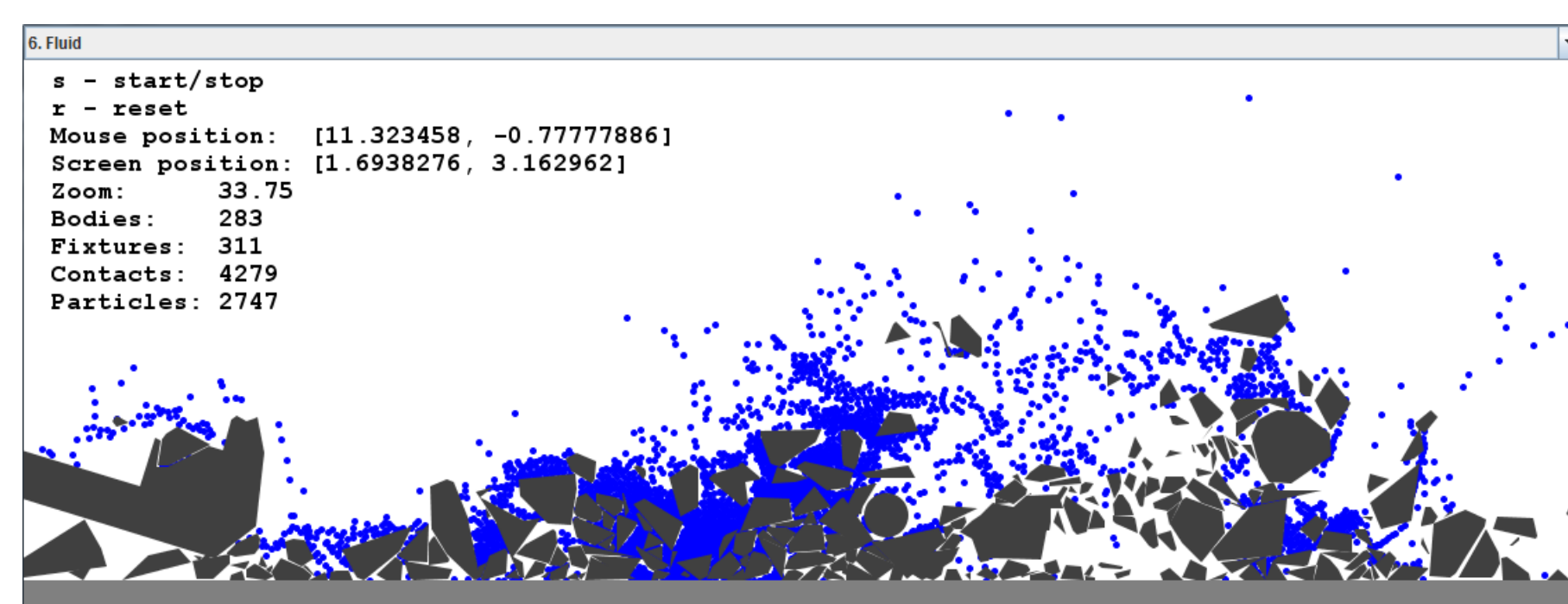
Konvexná dekompozícia

Kvôli optimalizácii spracovania kolízií je možné v pôvodnej verzii enginu definovať len tvary predmetov spĺňajúce definíciu konvexity a nekonvexné teleso je nutné z týchto útvarov zložiť. Na tento účel je aplikovaný mechanizmus konvexnej dekompozície. Tá je realizovaná trianguláciou polygónu a použitím Hertel-Mehlhornovho algoritmu odstraňujúceho hrany zachovávajúce konvexitu útvarov [2].



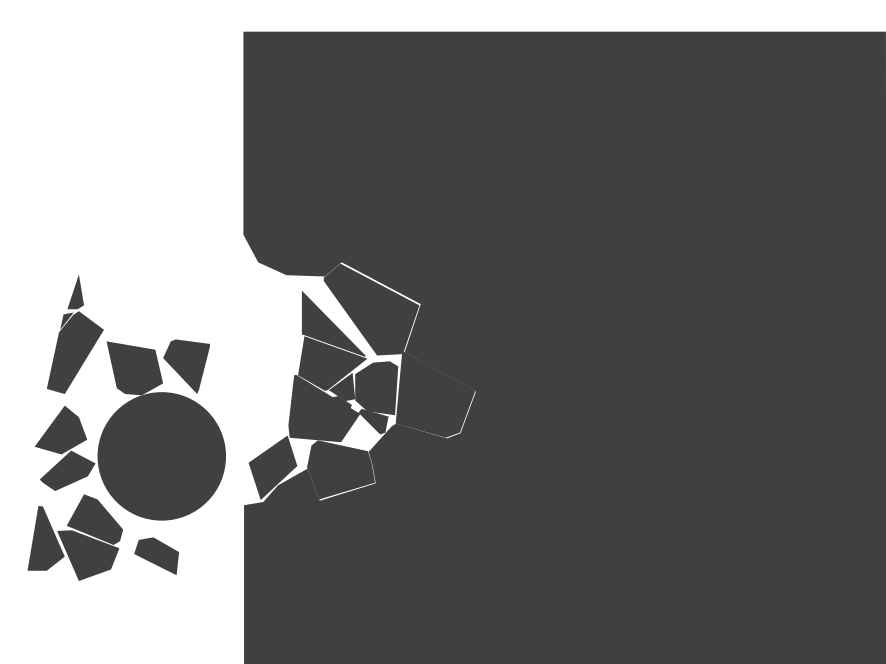
Grafické rozhranie

K dispozícii je jednoduché grafické rozhranie s testovacími scenármi simulujúcimi fraktúry objektov. Umožňuje výber testovacieho scenára a jeho interaktívne ovládanie.

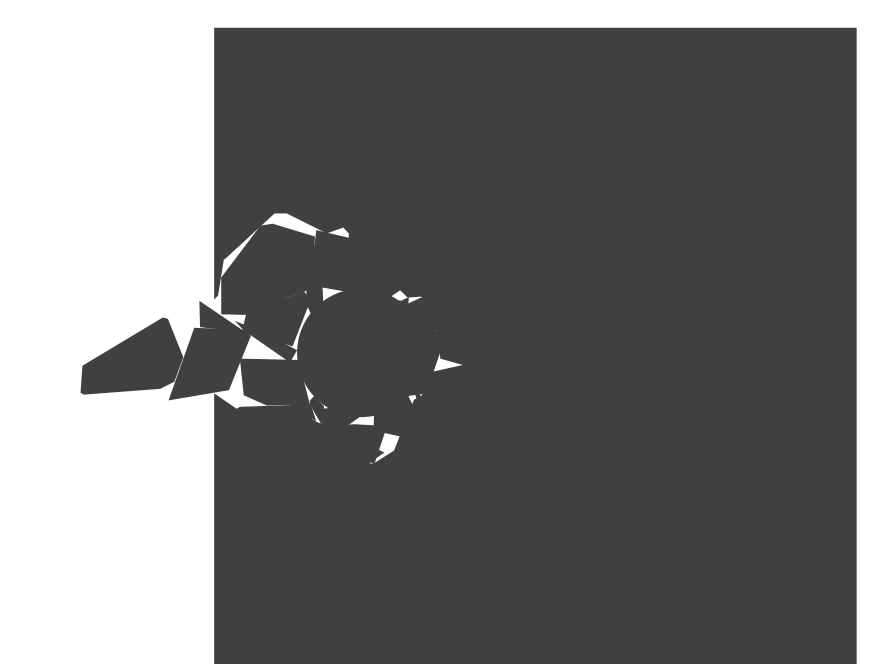
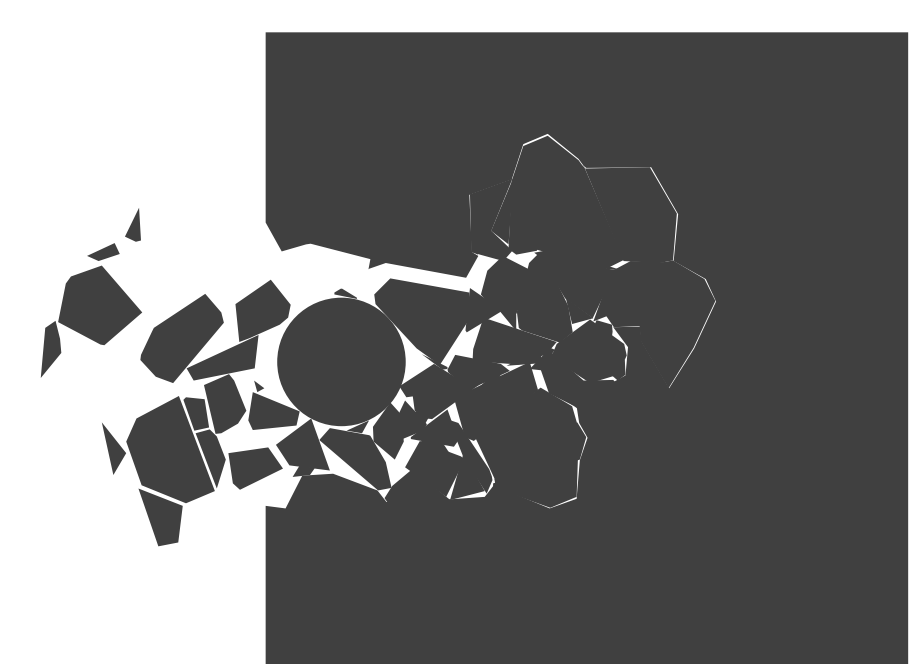


Výsledky

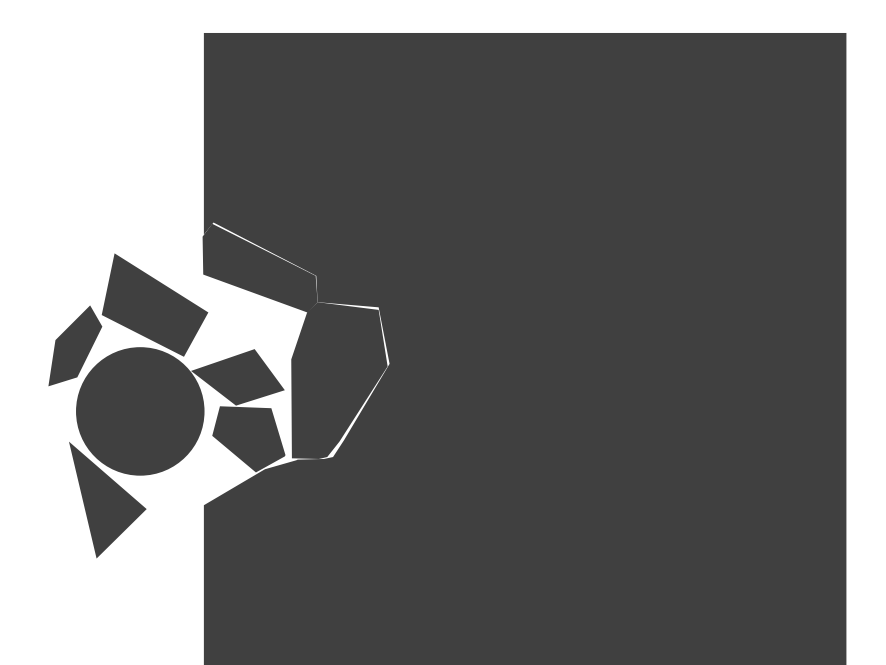
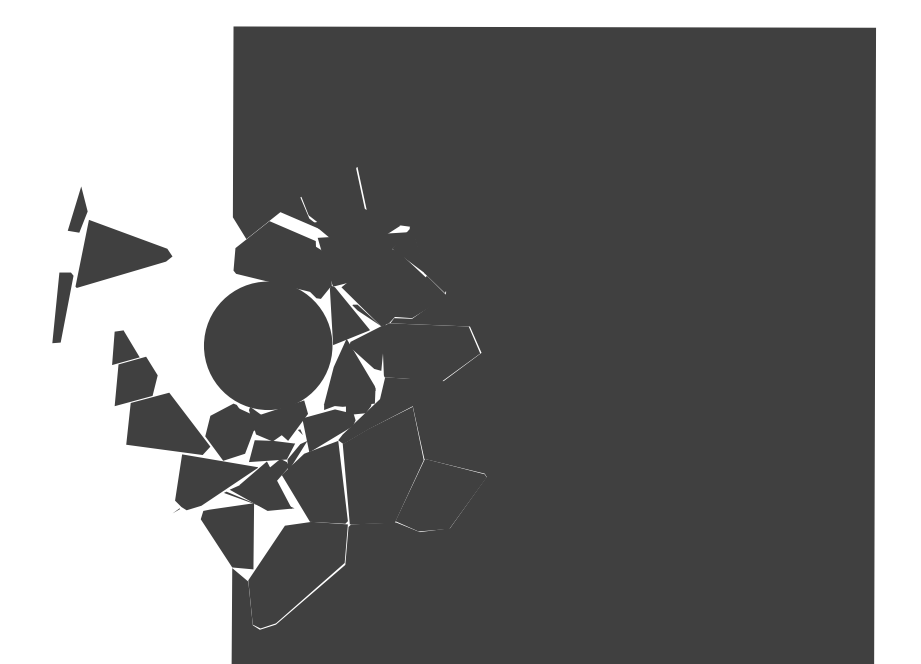
Materiály majú 3 atribúty: pevnosť, polomer účinnosti a mieru triedivosti. Výsledky testov triedení pri rôznom nastavení týchto atribútov vyzerajú nasledovne:



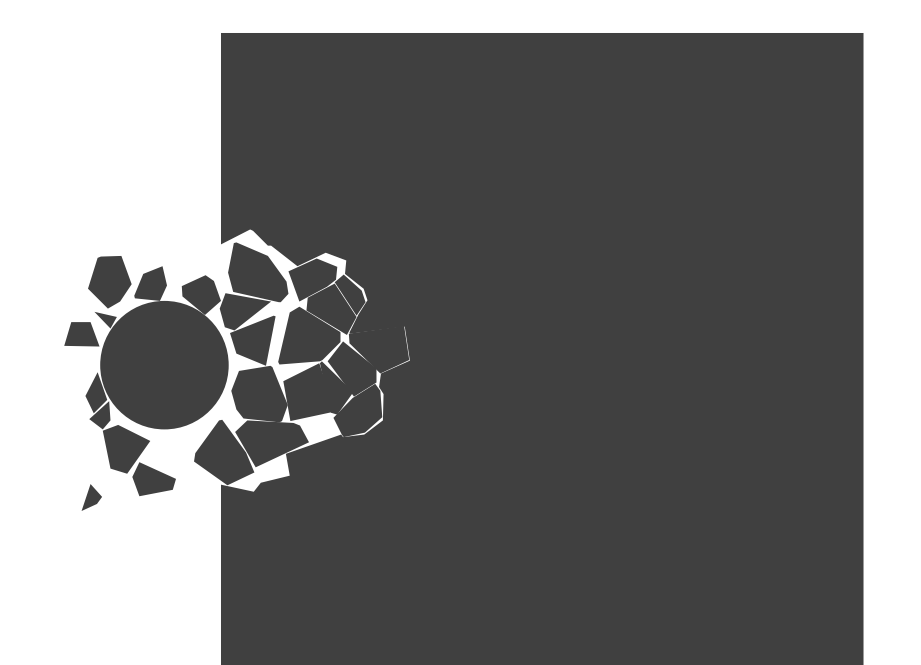
Správanie telies pri rôznej pevnosti materiálov



Správanie telies pri rôznom polomere účinnosti



Správanie telies pri rôznej miere triedivosti



Rozhranie

Definovanie nekonvexných telies:

```
PolygonFixture pol = new PolygonFixture(  
    new Vec2[] { ... }  
);  
body.createFixture(fixtureDef, pol);
```

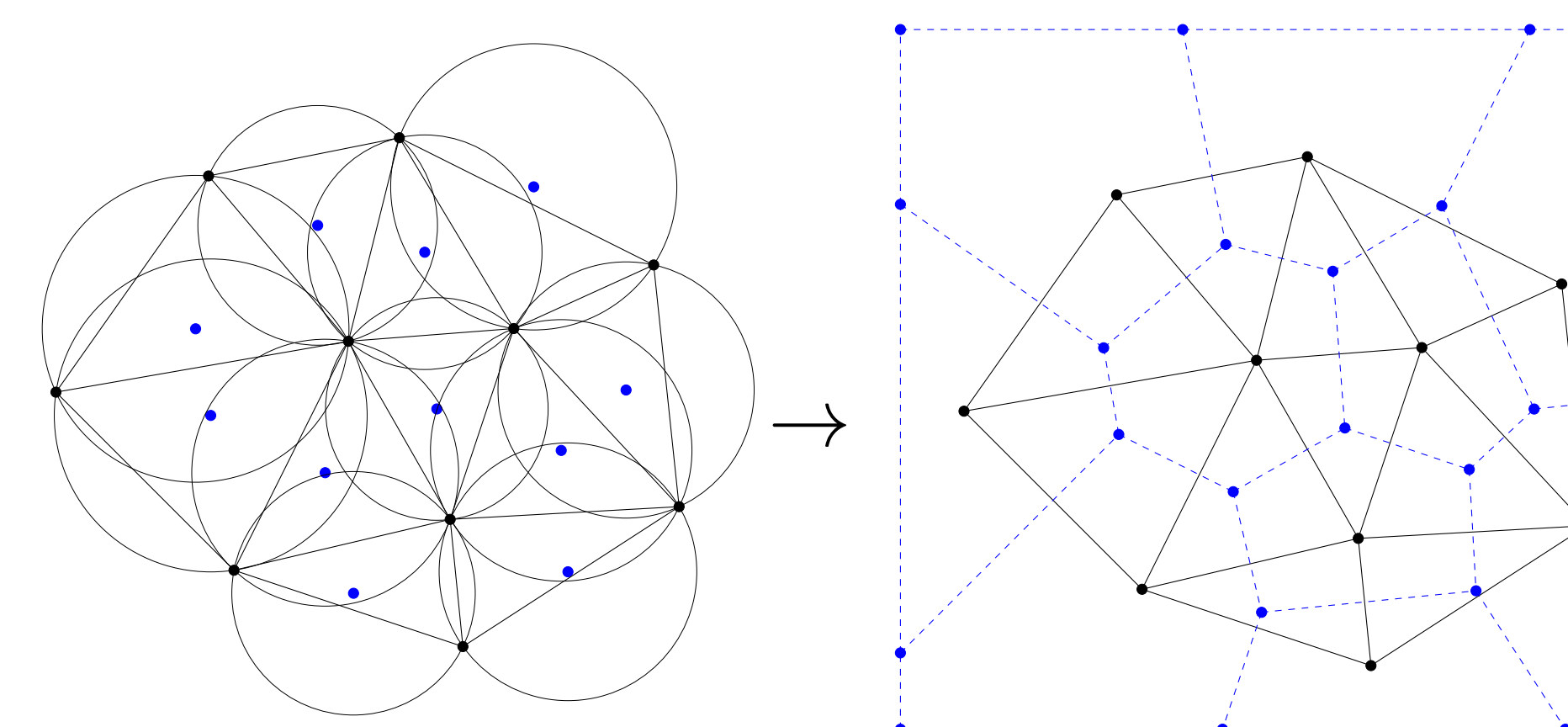
Aplikovanie triedivosti na predmet:

```
f.m material = Material.UNIFORM;
```

Materiálom je následne možné nastaviť vyššie spomenuté atribúty.

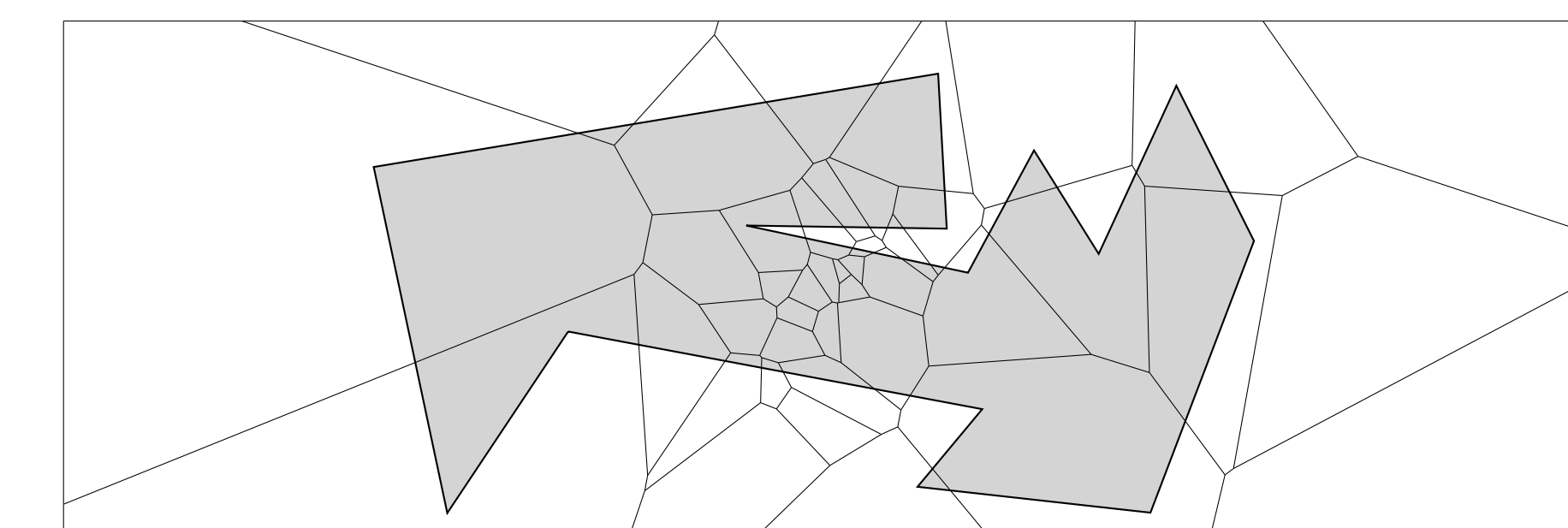
Voroného diagram

Na teseláciu telies sa využíva voroného diagram. Ten je implementovaný pomocou algoritmu otáčania hrán, ktorý v čase $\mathcal{O}(n \log n)$ počíta delaunayho trianguláciu, ktorá je následne prevedená na duálny graf - voroného diagram.



Fragmentácia a materiály

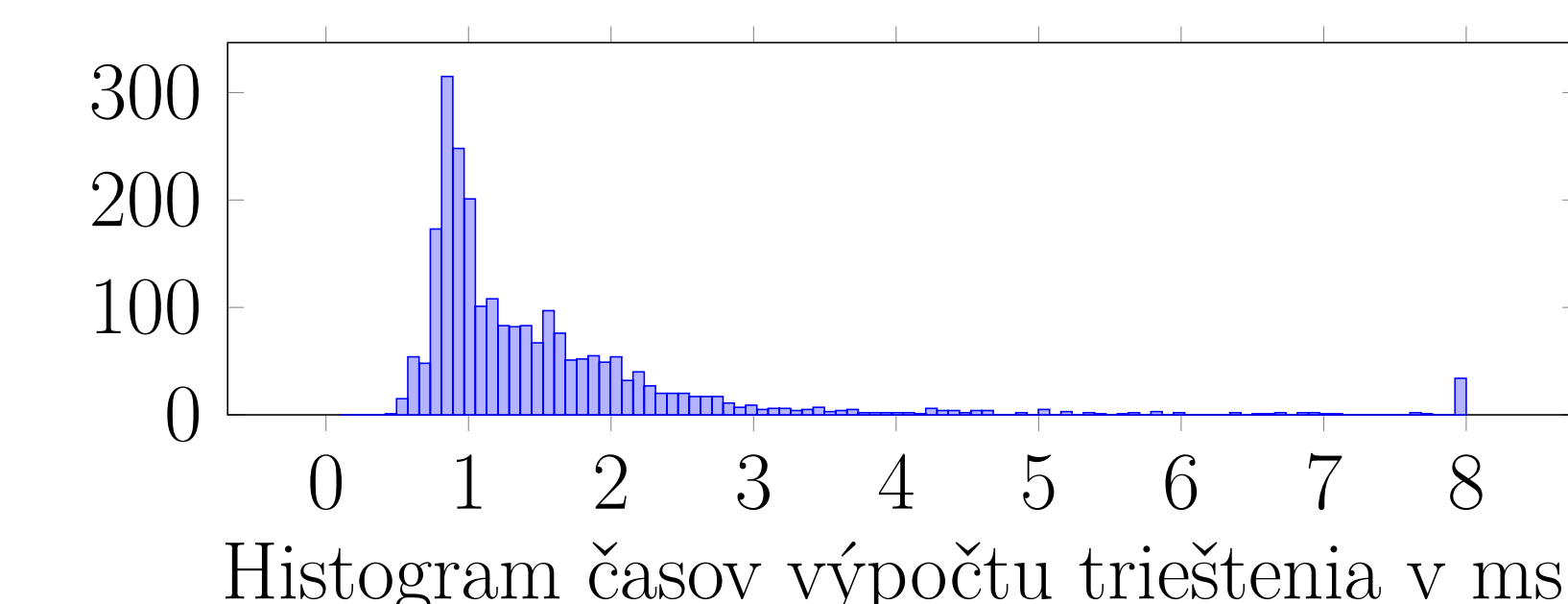
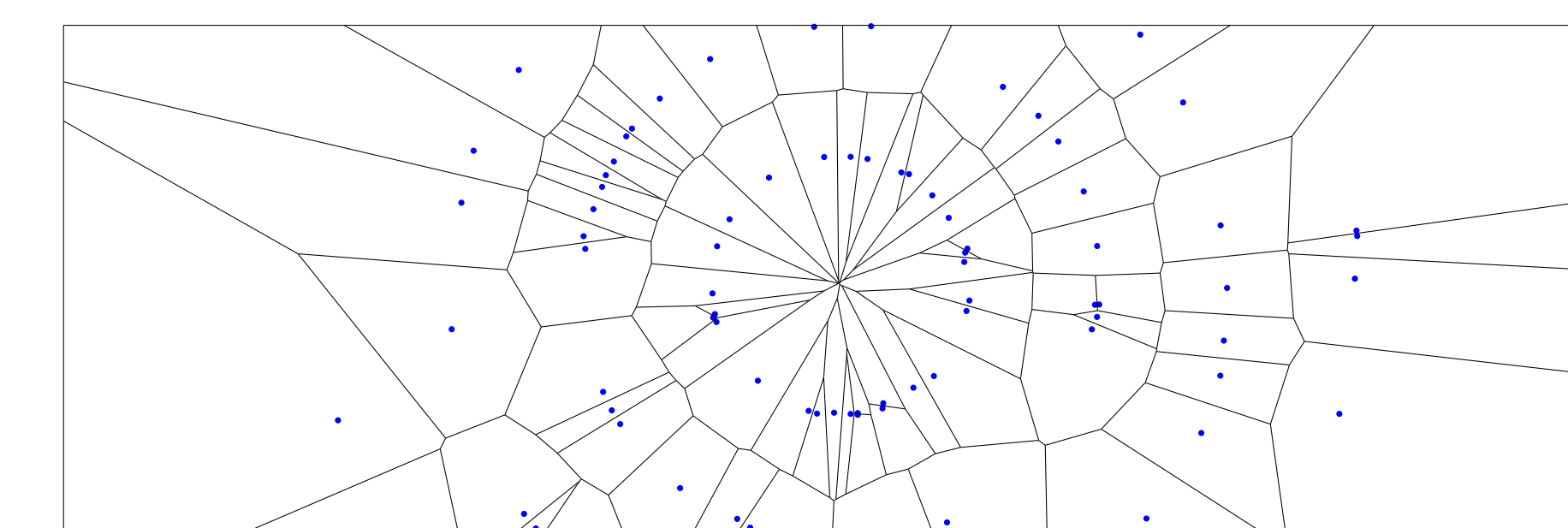
Software využíva metódu voroného fragmentácie - rozdelenia telesa pomocou voroného diagramu a následnom spracovaní a modifikovaní výsledných fragmentov.



Tvar úlomkov je možné definovať pomocou generátora ohnísk diagramu. Príklad:

$$\begin{aligned} \alpha &= p()2\pi && \text{(náhodný uhol)} \\ r_i &= s(i + p()c) && \text{(polomer)} \\ v_{i,k} &= (\sin \alpha r_i, \cos \alpha r_i) + d && 1 \leq i \leq n, i \in \mathbb{N} \end{aligned}$$

kde n určuje počet kružníc a c je miera vychýlenia.



Záver

V práci je možné nadväzovať rozšírením funkcionality ako podpora vkladania polygónov s dierami, explózie, tlakové vlny, deformácia, statická deštrukcia a iné. Práca poskytuje nástroje pre vývoj hier pre operačný systém *Android*.

Zoznam použitej literatúry

- [1] JBox2D. <http://www.jbox2d.org/>.
- [2] O'ROURKE, J. *Computational Gemoetry In C*. Second Edition. Cambridge University Press, 1997.