

EXAMENSARBETE Performance Optimisation of Collision Detection Algorithm in Particle Simulation**STUDENT** Johan Ravnborg**HANDLEDARE** Rikard Olajos (LTH)**EXAMINATOR** Michael Doggett (LTH)

Snabbare kollisionsdetektering i simulering av rörliga partiklar

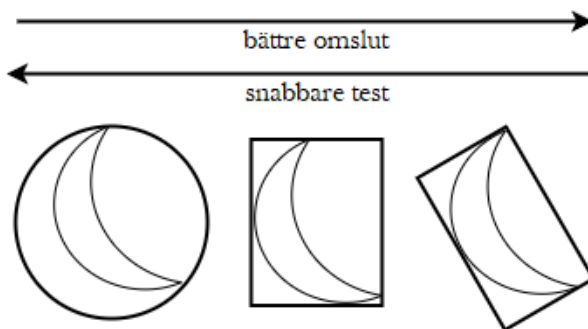
POPULÄRVETENSKAPLIG SAMMANFATTNING **Johan Ravnborg**

Snabb kollisionsdetektering påverkar allt ifrån molekylsimuleringar till datorgrafik. Detta arbete har stegvis förbättrat en enkel kollisionsdetekteringsalgoritm och jämfört hur det påverkat prestandan. I en simulering av partiklar som rör sig i två dimensioner såg vi stora förbättringar av att snabbt gruppera närliggande partiklar.

För att minska tiden vid kollisionsdetektering finns många tekniker. Ofta har objekt komplexa former som behöver längre tester - dessa sparar man tid på att undvika testa direkt. Roterade boxar är en vanlig form som tar längre tid att testa. För dessa tester genereras axlar med hjälp av boxarnas sidor, som deras hörn senare jämförs med hjälp av. Ett sätt att undvika dessa längre tester är genom att omsluta boxarna med cirklar, vars kollisioner är snabba att testa. Om två cirklar kolliderar görs sedan det längre testet mellan två roterade boxar. Ännu ett sätt att minska antalet tester på är med hjälp av trädstrukturer där närliggande objekt grupperas ihop, oftast i en stående omslutande box. Det finns träd där grupperingen är långsammare och försöker minska hur stora de omslutande boxarna blir vilket minskar hur många kollisionstester som måste göras. Det finns också snabbare, mindre exakta, algoritmer som upprepat delar upp objekten i hälften.

I mitt examensarbete har jag utgått ifrån en redan implementerad simulering. I denna rör sig en mängd partiklar mot deras gemensamma masscentrum för att generera kollisioner, och där kollisioner får partiklarna att studsa ifrån varandra. Programmet började med en enkel kollisionsdetektering som jag stegvis försökte förbättra. Dessa

förbättringssteg innefattade att generera axlarna till de roterade boxarna, att omsluta de roterade boxarna med cirklar, samt att gruppera objekten i två olika träd-strukturer, en som byggdes snabbare men mindre exakt, och en långsammare.



Körningstiderna mättes för de olika versionerna som testades, för varierande antal objekt.

Resultatet visade att de främsta förbättringsstegen vid höga antal objekt kom från att gruppera objekten i trädstrukturer. Framförallt trädstrukturen med snabba men mindre exakta byggalgoritmer presterade väl, 36 gånger snabbare än den bästa lösningen utan trädstruktur. Att omsluta objekten med cirklar gjorde programmet nästan 5 gånger snabbare än tidigare kollisionsdetekterings versionen som inte grupperades i träd.