

EXAMENSARBETE Highly Scalable Queues and Stacks with Elastic Relaxation**STUDENT** Kåre von Geijer**HANDLEDARE** Philippos Tsigas (Chalmers), Jonas Skeppstedt (LTH)**EXAMINATOR** Flavius Gruian (LTH)

Effektiva, flexibla och inexakta köer

POPULÄRVETENSKAPLIG SAMMANFATTNING Kåre von Geijer

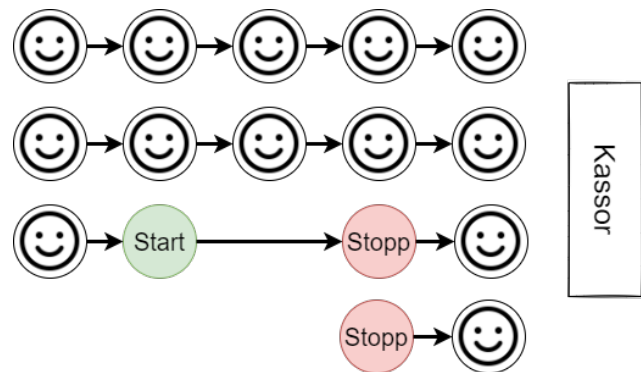
För att undvika flaskhalsar i program vill man ha effektiva och skalbara strukturer för att lagra och ordna data. Detta arbete har byggt vidare på två typer av effektiva men inexakta köer så att de flexibelt kan ändra sin precision vid behov.

Nya datorer blir effektivare varje år men nu blir de inte längre så mycket snabbare, utan får ofta istället fler kärnor. En kärna är som en egen minidator och detta gör att man kan köra allt mer instruktioner samtidigt i datorer. Men för att köra saker samtidigt måste man anpassa hur man skriver program.

Ett exempel är grundläggande datastrukturer som köer. De representerar ungefär en kö i mataffären där folk ställer sig i slutet av kön och kallas fram till kassan från början av kön. Men i vårt fall är det en kö av information istället för personer. När ett program kör på flera kärnor kan man se det som att det finns väldigt många kassor. Att bara ha en kö blir då långsamt då alla kassor måste tävla om att ropa fram den längst fram i kön hela tiden. Istället har vi ofta flera köer, kanske en till varje kassa, vilket gör det hela smidigare men inte längre lika rättvist (exakt) då köerna rör sig olika snabbt. Detta är i princip grundtanken med en tidigare publicerad artikel om effektiva men inexakta datastrukturer (tänk köer) som detta arbete bygger vidare på.

Detta arbete vidareutvecklar dessa datastrukturer och möjliggör dem att ändra hur inexakta de är under körning, vilket kan liknas vid att ändra antalet köer. På detta sätt kan man anpassa inexaktheten efter hand beroende på belastning, önskad precision, tillgängliga kärnor och så vidare för att få ett mer flexibelt program.

Vår lösning för kön illustreras i bilden nedan och kan liknas vid att ställa in start och stopp skyltar i olika köer när man vill öka respektive minska antalet. Dessa ställs in längst bak i köerna och flyttas fram allt eftersom andra köer rör sig framåt. Vi presenterar en lösning för andra typer av köer som bygger på att man håller koll på förändringar i en extra kö vid sidan av dem ursprungliga.



Våra resultat visar att det inte är stor skillnad i prestanda mellan våra mer flexibla köer och de ursprungliga om man inte tillåter antalet att ändras. Dessutom bevisar vi deras korrekthet och begränsar hur inexakta de är beroende på deras parametrar. Att de nya datastrukturerna inte är mycket långsammare än de vanliga, men nu är flexibla, gör att de kan öka totala prestandan över längre körningar om man utnyttjar deras flexibilitet.